SERVICIOS INTERNET EN LINUX

Ramón Mª Gómez Labrador
(ramon.gomez@eii.us.es)

Marzo 1.999

Temario.

Introducción. 1

Consideraciones generales. 1
Internet. 1
TCP/IP. 2
Servicio de nombres. 4

Configuración de la red. 6

Configuración básica. 6
Configuración posterior. 9

Servidor de hipertexto (WWW). 11

El lenguaje HTML. 11
El servidor Apache. 13
Configuración manual del Apache. 15
Configuración del Apache utilizando Comanche. 17
Seguridad. 17
Ejemplo. 21

Servidor para la transferencia de ficheros (FTP). 26

Configuración previa. 27
El servidor WU-FTPd. 27
Configuración del servidor WU-FTPd. 28
Ejemplo. 31

Servidor de correo electrónico. 35

El servidor Sendmail. 36
Configuración del Sendmail. 37
El servidor POP3. 40
Configuración del POP3. 40
Seguridad. 41
Netscape Messenger como cliente POP3. 42

Comunicación entre Linux y la red de Microsoft. 44

La utilidad SAMBA. 44
Configuración previa. 45
Configuración de SAMBA. 46
Seguridad. 51

Ejemplo. 51

Protección del servidor. 55

Protección de servicios con TCP-Wrappers. 55
Servidor representante. 56
Cortafuego. 60

Introducción.

Consideraciones generales.

Este curso está orientado a los administradores de sistemas que pretenden configurar aquellos servicios de comunicaciones para Internet que permiten ofrecer y acceder a la información de diversas maneras. Las notas de configuración y las aplicaciones que se verán en los próximos capítulos hacen referencia al sistema operativo Linux RedHat 5.2, sin embargo la mayoría de ellas son válidas para aquellos servicios instalados en ordenadores basados en cualquier otro tipo de sistema Unix.

No se pretende dar una descripción exhaustiva ni "a bajo nivel" de cada uno de los protocolos de comunicaciones y de las herramientas que los utilizan, sino una serie de ideas básicas que harán posible que el gestor de la máquina realice las siguientes operaciones:

- Configurar y gestionar los servicios de comunicaciones y sus recursos.
- Obtener una rápida información para solucionar los problemas más comunes.
- Aprender a manejar los clientes para cada uno de estos servicios.
- Establecer requisitos mínimos de seguridad.

Aunque todos los conceptos y ejemplos incluidos en este curso son de fácil comprensión para cualquier administrador de sistemas, éste deberá tener una mínima base en el tratamiento de máquinas Unix, en general, y Linux, en particular. Para obtener información complementaria, el gestor deberá referirse a la documentación suministrada por el propio sistema operativo.

Como complemento previo, recomendamos al lector la revisión del curso "Redes de Área Local y Sistemas Operativos en Red" del Secretariado de Formación de la Universidad de Sevilla.

Internet.

Internet es heredera directa de la red del Departamento de Defensa de Estados Unidos (ARPAnet) y de otras redes de investigación –fundamentalmente académicas– que se fueron uniendo a ésta.

A partir del año 1.983 –y coincidiendo con la adopción del protocolo de comunicaciones **TCP/IP**– ARPAnet se convierte en una de las subredes de **Internet**. Desde esta fecha Internet se basa en la interconexión de múltiples y muy diversas subredes, tanto redes de área local (LAN), como redes metropolitanas (MAN) o redes extensas (WAN).

La Red Nacional de I+D (**RedIRIS**) está conectada a Internet desde 1.991 y ofrece soporte a las universidades y centros de investigación españoles. El equivalente andaluz es la Red Informática y Científica de Andalucía (**RICA**).

La aparición durante la década de los 90 de una nueva generación de servicios y aplicaciones de comunicaciones y, sobre todo, de la "telaraña mundial" (*World-Wide Web* o simplemente WWW) ha hecho posible un crecimiento espectacular del número de subredes y de usuarios de Internet.

Como Internet no tiene un control centralizado, todo ordenador situado en "la Red" establece los servicios que necesita de ella y los que va a ofrecer al resto de la comunidad. Por lo tanto, se hace imprescindible que el administrador del sistema sea capaz de controlar y configurar una amplia variedad de estos servicios y de las utilidades que los gestionan.

TCP/IP.

El protocolo de comunicaciones **TCP/IP** (*Transmision Control Protocol/Internet Protocol*) permite la localización y comunicación de todo tipo de máquinas conectadas a Internet. TCP/IP es un protocolo basado en capas:

- La capa de red –equivalente al nivel 3 de la norma OSI–, que establece el camino óptimo que deben seguir los paquetes de información que comunican varias máquinas. Utiliza el protocolo **IP**.
- La capa de control de la transmisión –equivalente al nivel 4 de la norma OSI–, que permite establecer una comunicación segura con recuperación de datos en caso de error. Existen dos protocolos fundamentales:
  - el **TCP** propiamente dicho, que establece una comunicación síncrona entre los ordenadores; y
  - el **UDP** (*User Datagram Protocol*), que permite una comunicación asíncrona basada en paquetes denominados **datagramas**.

El protocolo de red establece un mecanismo basado en direcciones y nombres que localiza inequívocamente cada máquina conectada.

Una **dirección IP** es única para cada ordenador de la red y –en la versión actual del protocolo (IPv4)– consta de 4 bytes –con un rango de 0 a 255– separados por puntos. La versión IPv6 prevé la definición de un espacio de direcciones más amplio.

Las máquinas que se encuentren en la misma red tendrán una porción de su dirección IP que será común para todas ellas. Los números restantes permitirán identificarla inequívocamente dentro de dicha red. La siguiente tabla muestra las características fundamentales de las distintas clases de redes:

<table>
<tr><th>Clase</th><th>Byte 1</th><th>Byte 2</th><th>Byte 3</th><th>Byte 4</th></tr>
<tr><td>A</td><td><code>0</code> + nº de red (7 bits)
Rango: 1 – 126</td><td colspan="3">Dirección de máquina de 24 bits
Rango: 0.0.1 – 255.255.254</td></tr>
<tr><td>B</td><td colspan="2"><code>10</code> + nº de red de 14 bits
Rango: 128.1 – 191.254</td><td colspan="2">Dirección de máquina de 16 bits
Rango: 0.1 – 255.254</td></tr>
<tr><td>C</td><td colspan="3"><code>110</code> + nº de red de 21 bits
Rango: 192.0.1 – 223.255.254</td><td>Dir. máquina (8 bits)
Rango: 1 – 254</td></tr>
<tr><td>D</td><td colspan="3"><code>1110</code> + nº de red de más de 20 bits
Rango: 224.0.0.x – 254.255.255.x</td><td>Dir. máquina (menos de 8 bits)</td></tr>
</table>

Existen una serie de números de red reservados:

- 127, que identifica a la propia máquina.
- 192.168.x, que permite definir redes ocultas (intrarredes).

En cada red hay también dos números reservados:

- El menor de ellos se utiliza para indicar la dirección completa de la red.
- El último sirve como dirección de difusión para todas las máquinas de la red.

La red académica andaluza (RICA) es una red de clase B con dirección 150.214.0.0. Sin embargo, dentro de ella existen multitud de subredes de tipos C y D. Para ello se utiliza el concepto de máscara de red.

La **máscara de red** es el valor que, al realizar una operación lógica Y con la dirección de cualquier máquina, se obtiene la dirección de la red. Utilizando una máscara de red adecuada se consigue aislar el tráfico de las máquinas conectadas a un tramo común.

Ejemplo:

<table>
<tr><th colspan="2"><code>150.214.141.131</code></th></tr>
<tr><td>Dirección de la subred</td><td>Dirección dentro de la subred</td></tr>
</table>

Ejemplo:

Dirección de la máquina:	`150.214.141.131`
Máscara de red:	`255.255.255.0`
Dirección de la red:	`150.214.141.0`

La subred 150.214.141.0 actúa como una red de tipo C con hasta 254 máquinas conectadas.
La dirección 150.214.141.255 es la dirección de difusión de la red.
La dirección 150.214.141.1 suele utilizarse como dirección de la máquina que realiza las funciones de *encaminador* de la red.

En cada subred hay una máquina denominada ***encaminador*** (*router*), que determina el camino que deben tomar los paquetes que van dirigidos o que vienen de ordenadores de otras subredes.

Servicio de nombres.

El **servicio de nombres** asigna a cada dirección IP un nombre, aunque también puede asignarle varios **alias**. Tanto los nombres como los alias deben ser únicos en toda la red, para ello el servicio está gestionado localmente en cada zona geográfica y todo ordenador que va a conectarse a la red debe ser dado de alta en él, indicando su dirección IP, su nombre y sus posibles alias.

Un **nombre de red** consta de varios campos separados por puntos:

- El primero de ellos es el **nombre del ordenador** propiamente dicho, e identifica a las máquinas propias de una determinado organización.
- El resto de campos forman el **nombre del dominio** identificador de una empresa u organización. Un dominio está organizado de forma jerárquica y ascendentemente y debe tener un mínimo de 2 campos. El último campo indica el tipo de organización o el país al que pertenece.

La siguiente tabla muestra los distintos clases de dominios y algunos ejemplos con algunos dominios de cada tipo.

Países (2 letras)	es	España
	fr	Francia
	uk	Reino Unido
Otros dominios (más de 2 letras):	org	Organización
	com	Comercial
	int	Internacional

Ejemplo:

Algunos dominios	
Universidad de Sevilla	us.es
CPD de la U.S.	cpd.us.es
Fac. Informática y Estadísticas de la U.S.	fie.us.es
C.I.C.A.	cica.es
RedHat	redhat.com
Open Group	opengroup.org
W3 Consortium	w3.org

Ejemplo:

Nombre	Dominio	Nombre completo	Alias
casiopea	fie.us.es	casiopea.fie.us.es	www.fie.us.es ftp.fie.us.es

Todo ordenador debe conocer la dirección de la máquina que realiza la función de **servidor de nombres**. Por seguridad, los sistemas operativos permiten incluir direcciones de otros servidores de nombres que utiliza cuando el servidor principal está fuera de servicio. La siguiente tabla muestra las direcciones de los servidores de nombres que deben incluirse en los ordenadores de la Universidad de Sevilla conectados a Internet.

Servidor principal de la Universidad de Sevilla:	**150.214.186.69**
Servidor secundario (Rectorado):	**150.214.130.15**
Servidor del CICA (Reina Mercedes):	**150.214.4.34**

Configuración de la red.

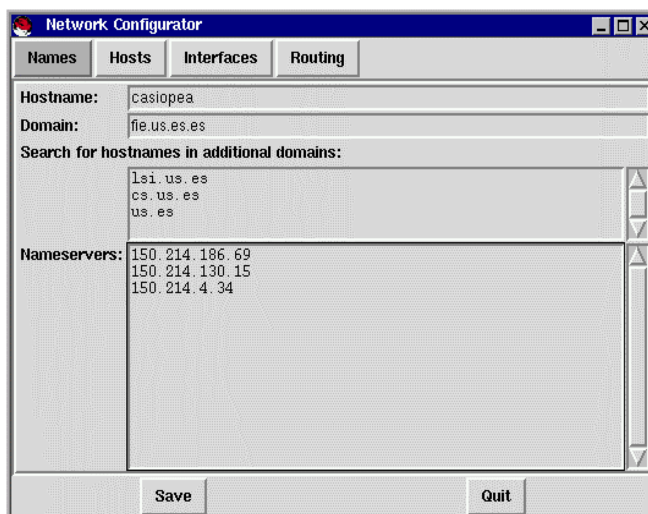
Configuración básica.

Antes de empezar a trabajar con la red, el administrador de la máquina debe configurarla correctamente, indicando:

- Tipo de controlador de red. El sistema operativo Linux está sufriendo un auge espectacular, por ello las nuevas versiones vienen mejor equipadas y soportan un mayor número de dispositivos. Sin embargo –debido a la filosofía aperturista del Linux– si el ordenador viene configurado con alguna tarjeta no reconocida por el sistema operativo, el administrador puede descargar de la red y compilar un módulo para dicho controlador (ver páginas de manual del mandato **`insmod`** para obtener más información sobre la inclusión de módulos).
- Dirección IP, nombre y posibles alias para la máquina.
- Dirección IP del *encaminador* (*router*) de la subred.
- Máscara de la subred.
- Dirección IP del servidor de nombre principal. Opcionalmente pueden indicarse también las direcciones de servidores secundarios.
- Opcionalmente pueden establecerse direcciones de otros ordenadores de interés para el correcto funcionamiento del sistema (servidor de correo electrónico, de ficheros NFS, de dominio NIS, de impresión, de arranque remoto, etc.).

La mayoría de estos parámetros suelen establecerse durante la instalación del sistema operativo, aunque el administrador (usuario `root`) puede modificarlos en cualquier momento editando los ficheros de configuración o utilizando las herramientas de gestión suministradas con el sistema operativo.

El gráfico adjunto presenta la ventana de ejecución del mandato **`netcfg`** (una de las utilidades del panel de control que aparece en la sesión X de `root`). Este programa permite ver y establecer los parámetros básicos del entorno de red. Para ello, se subdivide en 4 secciones principales, descritas en la siguiente tabla.



Network Configurator

Nombres:	Nombre y dominio del ordenador, direcciones de los servidores de nombres y otros dominios importantes.
Máquinas:	Asociación entre direcciones IP y nombres de máquinas.
Interfaces:	Configuración de las interfaces de red.
Rutas:	Caminos que deben seguir los paquetes de datos en función de su destino.

La tabla siguiente presenta una descripción detallada de algunos de los ficheros de configuración modificados por **`netcfg`**:

`/etc/hosts`	
Descripción:	Almacena la asociación entre dirección IP, nombre y alias de ordenadores conocidos.
Ámbito:	Todos los sistemas operativos Unix.
Formato:	*`DirecciónIP Nombre`* `[` *`Alias`* `... ]` `...`

Ejemplo:	`127.0.0.1        localhost`
	`150.214.141.131 casiopea  www      ftp`
	`150.214.141.104 antena    mailhost`
	`150.214.142.14  murillo`

`/etc/resolv.conf`	
Descripción:	Establece las bases para la resolución de nombres, indicando dominio del ordenador, dirección de los servidores de nombres y otros dominios de interés.
Ámbito:	Todos los Unix.

Formato:	**`Domain`** *`Dominio`*
	`Nameserver` *`DirecciónServidor`*
	`...`
	`[` **`Search`** *`OtroDominio`*
	`... ]`

Ejemplo:	`Domain fie.us.es`
	`Nameserver 150.214.186.69`
	`Nameserver 150.214.130.15`
	`Search lsi.us.es`
	`Search cs.us.es`
	`Search us.es`

`/etc/sysconfig/network`	
Descripción:	Establece los valores básicos de la red (nombre, dominio, dirección del *encaminador*, etc.
Ámbito:	Linux.

Formato:	*`Variable=Valor`* `...`

Ejemplo:	`NETWORKING=yes`
	`FORWARD_IPV4=no`
	`HOSTNAME=quintero.fie.us.es`
	`DOMAINNAME=fie.us.es`
	`GATEWAY=150.214.142.1`
	`GATEWAYDEV=eth0`

`/etc/nsswitch.conf`	
Descripción:	Indica el orden de búsqueda para ficheros de red.
Ámbito:	Todos los Unix modernos.
Formato:	*`TipoFichero TipoBúsqueda ...`* `...`
Tipos de búsqueda:	**`files`**: archivos locales. **`nis`**: NIS. **`nisplus`**: NIS+. **`dns`**: servicio de nombres.

Ejemplo:	`Passwd nis files`
	`Group  nis files`
	`Hosts  files dns nis`

`/etc/services`	
Descripción:	Indica el protocolo y el puerto utilizado por cada servicio de comunicaciones.
Ámbito:	Todos los Unix.
Formato:	*`Servicio Puerto/Protocolo`* `[` *`Alias ...`* `]` `...`

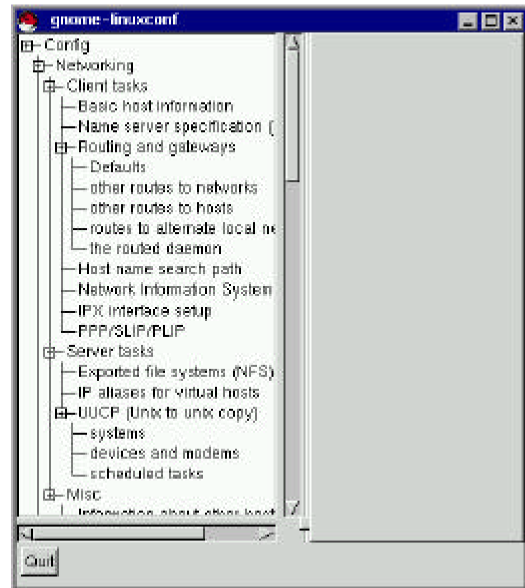
Ejemplo:	`echo     7/tcp`
	`echo     7/udp`
	`ftp-data 20/tcp`
	`ftp      21/tcp`
	`telnet   23/tcp`
	`smtp     25/tcp  mail`
	`http     80/tcp`
	`pop-3    110/tcp          # PostOffice V.3`

`/etc/inetd.conf`	
Descripción:	Fichero de configuración del programa servidor de comunicaciones **`inetd`**.
Ámbito:	Todos los Unix.
Formato:	*`Servicio Tipo Protocolo Espera Usuario Servidor Args`* `...`

Ejemplo:	`#echo  stream tcp nowait root internal`
	`#echo  dgram  udp wait   root internal`
	`ftp    stream tcp nowait root /usr/sbin/tcpd  in.ftpd -l –a`
	`telnet stream tcp nowait root /usr/sbin/tcpd  in.telnetd`
	`pop-3  stream tcp nowait root /usr/sbin/tcpd  ipop3d`

Los usuarios administradores de un ordenador con el sistema operativo Linux de RedHat versión 5.2 pueden realizar la mayoría de las operaciones de gestión utilizando el programa **`linuxconf`**. El siguiente gráfico muestra la pantalla principal de esta utilidad.

GNOME Linuxconf está estructurado en forma de árbol, con varias ramas principales –una de ellas es la relativa a la configuración de la red (*Networking*)– y unas opciones que despliegan pantallas de configuración en la zona derecha de la ventana.



Linuxconf

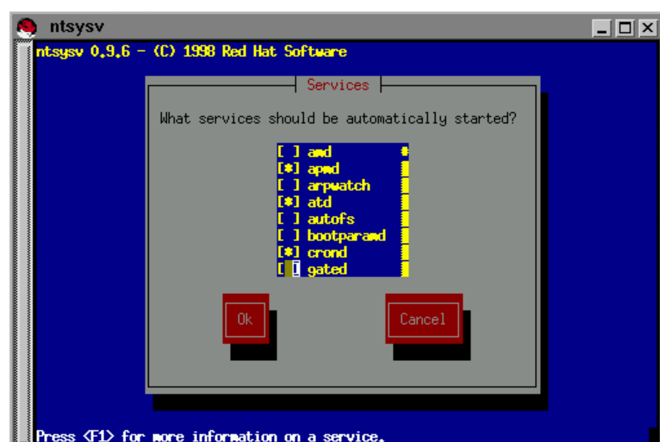
Las tareas para la configuración básica de la red se encuentran en el menú *Client tasks* de la rama *Networking*. En él pueden añadirse:

- Información básica (nombre, dirección IP, etc.).
- Servidores de nombres.
- Rutas específicas.
- Preparar del ordenador como un *encaminador.*
- Establecer el orden de búsqueda de ficheros.
- Configurar como servidor o cliente de los Servicios de Información de la Red (NIS o NIS+).
- Conexión a servidores NetWare (IPX).
- Conexión punto a punto, mediante puerto serie o mediante puerto paralelo.

Configuración posterior.

Una vez establecidos los parámetros básicos para la conexión a la red, el administrador deberá continuar la configuración de los siguientes aspectos:

- Indicar qué procesos van a activarse durante el proceso de encendido del ordenador, o sea, establecer los servicios de red que van a ser ofrecidos por el sistema.
- Realizar la configuración óptica para cada uno de dichos servicios.
- Preparar las políticas de seguridad para el acceso al ordenador.



ntsysv

Con respecto al primer punto, el programa de instalación del sistema

permite establecer una lista de procesos que se activarán al iniciar el ordenador. Sin embargo, el administrador puede modificar dicha lista "a mano" –editando los ficheros de configuración situados en el subdirectorio `/etc/rc.d`, o en sus subdirectorios– o ejecutando las herramientas de gestión ofrecidas por el sistema operativo. El gráfico anterior muestra la salida del mandato **`ntsysv`**.

Los otros dos últimos se estudian con detalle en el resto de los capítulos de este curso.

Servidor de hipertexto (WWW).

El servicio de hipertexto a contribuido en gran parte al auge de Internet, ya que permite ofrecer información mediante documentos que incluyen texto, gráficos, datos y enlaces o vínculos a otros documentos –que pueden estar almacenados en otros ordenadores–, creando una especie de *telaraña global*, (*World–Wide Web*). Para ver el contenido de dichos documentos debe utilizarse un programa especial denominado ***navegador***.

Este concepto fue creado por Tim Verners-Lee en el Centro Europeo para Física de Partículas (CERN), como un mecanismo para la comunicación e intercambio de información entre los investigadores del centro.

La WWW se basa en:

- Un protocolo de transporte conocido como HTTP (*Hipertext Transfer Protocol*).
- Un lenguaje que permite crear documentos conocido como HTML (*Hipertext Makeup Language*).
- Una estructura cliente/servidor, donde la información se publica en el servidor y el programa navegador (cliente) muestra los datos según el formato indicado en el documento.

Desde su creación, tanto el HTTP como el HTML han evolucionado rápidamente, incluyendo una serie de mejoras que permiten realizar documentos más comprensibles y cómodos de leer. También han aparecido nuevos lenguajes para una comunicación más interactiva entre el servidor y el usuario (CGI, *aplets* Java) y que contribuyen a crear estilos de presentación (CSS), a crear documentos dinámicos (XML) o a generar espacios de realidad virtual (VRML).

En los navegadores es en donde ha existido una evolución más rápida, ya que actualmente pueden utilizar varios protocolos e incluso ejecutar aplicaciones. Así, un mismo programa puede usarse para ver documentos, para transferir ficheros o para leer el correo electrónico.

El lenguaje HTML.

Un documento escrito en HTML es un fichero de texto, que consta de dos partes fundamentales: cabecera y cuerpo. Cada una de estas partes contiene una serie de palabras reservadas que realizan una determinada función interpretable por el navegador. Dichas palabras reservadas –y sus posibles parámetros– están encerradas entre símbolos de menor (<) y mayor (>). Para indicar el final de una acción, ésta suele ir precedida por el símbolo de división (/). El formato es:

```
<palabra [ parámetro=valor ... ] > texto </palabra>
```

El **Localizador de Recursos Uniforme** (URL, *Uniform Resource Locator*) es una norma que identifica la dirección de cualquier documento o fichero publicado en la WWW. Por lo tanto, un vínculo de hipertexto se especifica mediante un URL. El formato del URL es:

`protocolo://máquina:puerto/documento`

Ejemplo:

URLs.

```
http://www.fie.us.es/~ramon/articulos/malamente.html
http://murillo.fie.us.es:8080/
ftp://ftp.fie.us.es/pub/WWW/
```

La cabecera de un documento HTML puede incluir los siguientes datos:

- Título del documento.
- Tipo de codificación del documento (lenguaje, mapa de caracteres, etc.).
- Datos del autor (nombre, dirección de correo electrónico, etc.).
- Otros documentos relacionados (índice, hojas de estilos, *aplets*, etc.).

El cuerpo incluye el documento propiamente dicho. Las características principales son:

- Cabeceras.
- Párrafos.
- Listas.
- Tablas.
- Gráficos.
- Vínculos a otros documentos o a partes del propio documento.
- Formularios.
- Pequeñas aplicaciones o porciones de código.

Ejemplo:

Un documento HTML.

```
<HTML>
<HEAD>
  <TITLE>Título del documento</TITLE>
</HEAD>
<BODY BGCOLOR=#004000 TEXT=#FFFFFF>
  <H1>Hola.</H1>
  <P ALIGN=”CENTER”>Esto es un ejemplo de documento HTML.</P>
</BODY>
</HTML>
```

Las páginas HTML eran –en principio– estáticas. El usuario podía navegar por la telaraña, pero sólo recibía información, sin ningún tipo de interactuación. Los **programas CGI** (*Common Gateway Interface*) fueron el primer paso para permitir una comunicación básica entre el usuario y el servidor.

Las CGI son programas –normalmente guiones– que toman datos de un formulario o del navegador, ejecutan las operaciones en el ordenador del servidor y generan páginas HTML. Las CGI permiten generar páginas poco interactivas y requieren el mantenimiento de una seguridad alta.

El lenguaje Java puede usarse para incrustar pequeños programas –denominados **aplets**– en las páginas HTML. Este lenguaje es independiente respecto a la plataforma, con lo que pueden crearse aplicaciones que se ejecuten en una gran variedad de ordenadores y sistemas operativos sin tener que modificar el código. Paralelamente al desarrollo del Java ha aparecido una versión reducida del lenguaje conocida como **JavaScript**, que permite incluir el código en el propio documento HTML.

Resulta importante destacar que tanto los *aplets* Java como los guiones JavaScript se ejecutan en el ordenador del cliente, pero sólo si el navegador puede interpretar su código. Todos los clientes modernos soportan la ejecución de ambos tipos de programas.

Una nueva posibilidad la ofrecen los servidor de hipertextos que soportan las **servlets**. Éstas son aplicaciones que se ejecutan parcialmente en la máquina del servidor y parcialmente en el ordenador del cliente. Esta capacidad no aparece en todos los servidores HTTP y requieren una gestión apropiada de la seguridad.

El servidor Apache.

Apache es uno de los servidores para hipertextos de dominio público más utilizados y está incluido en la distribución del Linux de RedHat.

Existen dos maneras no excluyentes para configurar este programa:

- Editar los ficheros de configuración.
- Utilizar la herramienta **Comanche**.

Antes de comenzar a preparar la configuración del servidor, el administrador deberá tener en cuenta los siguientes aspectos:

Tipo de servidor:	Apache puede arrancarse como un servicio independiente (recomendado) o integrado dentro del `inetd` (como el Telnet).
Propietario y grupo:	Crear –si es necesario– el grupo y la cuenta del *usuario ficticio* que ejecuta las peticiones al servidor. Por motivos de seguridad, **no es recomendable que el usuario `root` –ni cualquier otro administrador– sea propietario de los procesos del servidor.**

Directorios:	Deberán crearse los directorios donde se encontrarán las páginas de hipertexto, los gráficos e iconos más comunes, los programas ejecutables (CGI, *aplets*, etc.) y los históricos de accesos al servidor. Establecer los permisos adecuados en cada uno de ellos.
Informes e históricos:	Establecer parámetros sobre el tipo de información que será almacenada en relación con los accesos al servidor. El administrador deberá revisar estos ficheros para corregir parámetros, prevenir intrusiones o evitar errores en los accesos al servidor.
Política de accesos:	Creación de usuarios y grupos propios del servidor para acceder a información reservada, posibilidad de mostrar el contenido de los directorios, permitir que los usuarios normales de la máquina puedan publicar páginas personales o ejecutar programas, restringir o permitir el acceso desde ordenadores o dominios específicos.
Otros programas:	Compilar y configurar otros programas que pueden ayudar a la gestión del servidor o que mejoran sus capacidades de presentación (contadores, servidores de *aplets*, interfaces para acceso a bases de datos, creación de copias de seguridad de los datos, etc.). El propio servidor Apache viene equipado con varios módulos de ampliación –que permiten generar estadísticas, compatibilidad con PERL, etc.– y que pueden ser añadidos durante el proceso de configuración.
Servidores virtuales:	Configuración de otros *servidores virtuales* que pueden definirse en el mismo ordenador. Especificar un nombre ficticio y una configuración completa para cada uno de ellos como si estuviesen localizados en máquinas diferentes.

Después de preparar una configuración inicial, el gestor del servidor de hipertextos deberá iniciar los procesos que lo activan. Cada vez que se modifica cualquier parámetro de configuración, deberá de procederse con la parada y al rearranque del servicio correspondiente (conocido como **`httpd`**).

Para completar la configuración, el administrador del sistema deberá realizar las siguientes operaciones:

- Ejecutar la orden **`ntsysv`** para incluir en el proceso de arranque del ordenador una referencia al servidor de hipertextos (servicio **`httpd`**). Otro procedimiento para lograr el mismo efecto es añadir el guión adecuado en el directorio para la configuración de los niveles de ejecución de la máquina.

- Indicar en el fichero de servicios de red (`/etc/services`) el puerto y el protocolo utilizado

por el **httpd**. Puede establecerse más de un puerto cuando se configuran servidores virtuales o cuando se incluyen otras utilidades de red complementarias (SSL, *servlets*, etc.).

- Si se ha preparado una configuración del **httpd** basada en el **inetd**, deberá añadirse la línea correspondiente en el fichero `/etc/inetd.conf`.

Los dos próximos apartados ayudarán a comprender cada aspecto relativo a la configuración del servidor de hipertextos. El administrador del sistema puede obtener mayor información dirigiéndose a la documentación suministrada con el servidor Apache o en la dirección de Internet http://www.apache.org/.

Configuración manual del Apache.

A partir de la revisión 1.3.4 del servidor Apache sólo se utiliza un único fichero de configuración (`httpd.conf`), para evitar posibles redundancias o confusiones. Sin embargo, la versión 1.3.3, suministrada con el Linux de RedHat 5.2, conserva aún tres ficheros de configuración:

httpd.conf:	Datos de control del servidor.
access.conf:	Datos para el control de accesos.
srm.conf:	Datos sobre especificación de ficheros.

Estos ficheros cumplen las siguientes características:

- Constan de una serie de directivas de configuración. Se clasifican en:
  - Simples: una directiva por línea, con el siguiente formato.

    ```
    Directiva Parámetro ...
    ```

  - Complejas: incluyen una o varias directivas, tanto simples como complejas. Su formato es el siguiente:

    ```
    <Directiva Parámetro ...>
          Directivas ...
          ...
    </Directiva>
    ```

- Las líneas con comentarios comienzan con el símbolo almohadilla (#).
- Las expresiones regulares suelen ir precedidas por el símbolo tilde (~).

Adicionalmente se distribuye otro fichero de configuración (`mime.types`) que permite especificar los tipos de documentos que serán suministrados por el **httpd**.

La siguiente tabla muestra algunas de las directivas más comunes para configurar un servidor Apache.

Directiva	Comentario:
LoadModule	Carga el módulo correspondiente. Cada módulo habilita una serie de directivas.
AddModule	Establecimiento del orden correcto de carga de módulos.
ServerType	Tipo de ejecución del servidor, independiente o basada en el **inetd**.
ServerName	Nombre completo del servidor.
Port	Puerto TCP gestionado por el servidor.
HostnameLookups	Búsqueda de clientes por nombre o por dirección IP. La búsqueda por nombres ralentiza la respuesta del **httpd**.
User	Usuario propietario de los procesos del servidor (debe ser un usuario ficticio).
Group	Grupo propietario de los procesos del servidor (debe ser un grupo ficticio).
ServerAdmin	Dirección de correo del administrador del servidor de hipertextos.
ServerRoot	Directorio de configuración.
ErrorLog	Fichero histórico de errores (referido a ServerRoot).
CustomLog	Otros ficheros históricos (referido a ServerRoot).
KeepAlive	Habilita o deshabilita el uso de conexiones persistentes.
StartServers	Nº de procesos servidores que deben arrancarse. Depende de la potencia del ordenador.
MaxClients	Nº máximo de procesos servidores en ejecución. También depende del tipo de ordenador utilizado.
Listen	Permite la escucha de otros puertos para crear servidores virtuales.
<VirtualHost>	Directivas de configuración de un servidor virtual.
<Directory>	Directivas de configuración para accesos a directorios.
<Location>	Directivas de configuración de servicios asociados a URLs.
DocumentRoot	Directorio donde se encuentran los documentos principales del servidor.
UserDir	Directorio de los documentos personales de los usuarios.
DirectoryIndex	Archivo o programa que contiene el índice o la página principal de un dierectorio.
AddIcon	Incluye iconos que permiten identificar tipos de archivos.

`Alias`	Asocia nombres (alias) a directorios
`ScriptAlias`	Indica los alias para directorios que incluyen programas.
`Redirect`	Indica a los clientes la dirección de documentos movidos de lugar.
`ErrorDocument`	Permite diseñar documentos que gestionan errores de acceso al servidor.

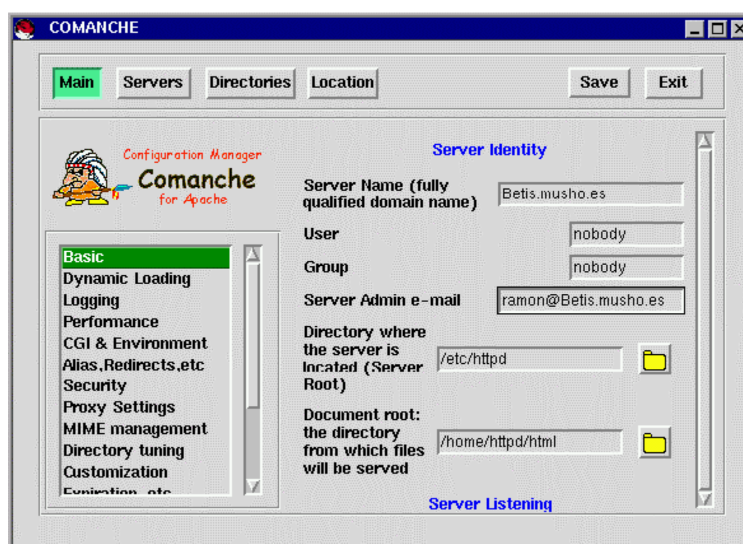
Configuración del Apache utilizando Comanche.

Comanche es una interfaz ejecutable desde Xwindows, que permite configurar de un modo gráfico y cómodo el servidor Apache.

Este programa consta de cuatro secciones principales, cada una de ellas subdividida en grupos de directivas de configuración. El gráfico contiguo muestra el grupo de parámetros básico (*Basic*) de la sección principal (*Main*).

Este programa permite establecer una configuración equivalente a la descrita en el apartado anterior.

Las secciones en las que se divide el programa **Comanche** son:



Comanche.

Principal (*Main*):	Directivas principales del servidor.
Servidores (*Servers*):	Directivas para servidores virtuales (`VirtualHost`).
Directorios (*Directories*):	Directivas para acceso a directorios (`Directory`).
Localización (*Locations*):	Directivas para localización de servicios (`Location`).

Seguridad.

Los archivos y directorios que forman parte del programa servidor de hipertextos sólo deben ser accesibles por los usuarios autorizados. Sólo aquellos directorios con información pública o propios de aquellas usuarios que pueden publicar datos, deben ser accesibles por el usuario ejecutor del **`httpd`**.

En ningún caso, un usuario deberá tener permiso para modificar los datos de otro usuario o la información principal del servidor.

En los próximos párrafos se incluyen las recomendaciones necesarias para incrementar la seguridad en el servidor. Cada uno de ellos irá acompañado con una o varias tablas que incluyen los mandatos que permiten realizar las funciones descritas. Cada línea puede ir precedida por un uno de los siguientes "puntos indicativos":

- #: mandatos ejecutados por el usuario administrador (`root`).
- $: mandatos ejecutados por otro usuario.

El gestor de la máquina debe ser el propietario de los directorios y de los archivos ejecutables, históricos y de configuración del Apache.

```
# cd DirectorioWeb
# chown root     . bin conf logs
# chgrp GrupoWeb . bin conf logs
# chmod 750      . bin conf logs
```

Los ficheros y directorios con información global del servidor deben ser accesibles sólo para lectura por el usuario propietario de los procesos del servidor (cláusula `User`). Sólo en casos especiales –como en los archivos históricos o en algunos documentos generados por programas CGI– éste usuario tendrá permisos de escritura.

```
# chown User  logs/*
# chgrp Group logs/*
# chmod 640   logs/*
```

Los ficheros y directorios que contienen información general del servidor –incluidos en el directorio indicado en la cláusula `DocumentRoot`–, deberán tener permiso de lectura para el grupo ejecutor de los procesos.

```
# chown Propietario DocumentRoot
# chgrp Group       DocumentRoot
# chmod 2750        DocumentRoot
$ chmod 640  FicheroWeb
$ chmod 2750 DirectorioWeb
```

Si los usuarios del ordenador también pueden publicar datos en el servidor, deberán tener en sus cuentas un directorio que coincida con el indicado en la cláusula `UserDir` y un fichero dentro de dicho directorio con el nombre indicado en la cláusula `DirectoryIndex`. Todos los ficheros y directorios publicados deben tener permiso de lectura par el grupo correspondiente a la cláusula `Group`.

```
$ cd
$ mkdir UserDir
$ chmod 711 .
$ chmod 755 UserDir
$ vi UserDir/DirectoryIndex
$ chmdo 644 UserDir/DirectoryIndex
```

Los programas CGI deben ser ejecutables por el usuario indicado por la cláusula `User` o por el grupo de la cláusula `Group`. Sin embargo, no es conveniente que dicho usuario tenga permiso de modificación en estos ficheros. El usuario `root` sólo será propietario de ficheros ejecutables en casos estrictamente necesarios.

```
# chown root    ScriptAlias
# chgrp Group   ScriptAlias
# chmod 2750    ScriptAlias
# chown Usuario ScriptAlias/FicheroCGI
# chmod 750     ScriptAlias/FicheroCGI
```

El servidor Apache incluye también varias directivas que permiten restringir o verificar el acceso a determinados documentos.

Las subdirectivas incluidas en las cláusulas **<Directory>**, **<Files>** y **<Location>** –y las equivalentes para expresiones regulares **<DirectoryMatch>**, **<FilesMatch>** y **<LocationMatch>**– establecen los permisos básicos para acceder a directorios, ficheros y URLs específicos, respectivamente. La siguiente tabla muestra las directivas permitidas.

Directiva	Comentario:
`Options`	Opciones de acceso. • `All`: todas excepto `MultiViews` (valor por omisión). • `ExecCGI`: se permite la ejecución de programas CGI. • `FollowSymLinks`: permite seguir enlaces simbólicos (ignorado en `<Location>`. • `Includes`: permite incluir ficheros y programas en documentos HTML. • `IncludesNOEXEC`: permite incluir ficheros, pero no programas. • `Indexes`: genera automáticamente un listado del directorio si en él no existe el fichero especificado en la directiva `DirectoryIndex`. • `MultiViews`: soporta la negociación de contenidos especificada en el protocolo HTTP/1.1. • `SymLinksIfOwnerMatch`: se siguen los enlaces simbólicos si el origen y el destino del enlace son del mismo usuario (ignorado en `<Location>`.
`AllowOverride`	Indica qué conjunto de opciones pueden solaparse mediante el fichero para la configuración de accesos al directorio (`.htaccess`).
`AuthConfig`	Configuración para autentificar usuarios con permiso de acceso.
<Limit>	Directivas aplicadas a determinados métodos de acceso del protocolo HTTP.
`Order`	Orden de preferencia para las cláusulas de permito y denegación.
`Allow`	Permite el acceso a los nombres, direcciones IP o dominios indicados.
`Deny`	Deniega el acceso a los nombres, direcciones IP o dominios especificados.
`Require`	Indique qué usuarios o grupos tienen permiso de acceso.

El orden para el procesamiento de los permsisos es:

1. Cláusulas de **<Directory>** sin expresiones regulares y las opciones solapadas por el fichero de accesos al directorio (.htaccess).
2. Cláusulas de **<DirectoryMatch>** y **<Directory>** con expresiones regulares.
3. Cláusulas de **<Files>** y **<FilesMatch>**.
4. Cláusulas de **<Location>** y **<LocationMatch>**.
5. Las directivas anteriores incluidas en la sección **<VirtualHost>** se aplican en último lugar.

La autentificación de usuarios y grupos se realiza a través de una serie de ficheros que deben estar fuera del árbol de directorios de documentos y únicamente con permisos de lectura para el usuario ejecutor de los procesos del servidor (directiva User).

Las siguientes tablas muestran los formatos de los ficheros típicos para usuarios y grupos y las directivas que los gestionan.

Usuarios:	*Usuario*:*Clave*
Grupos:	*Grupo*: *Usuario1 Usuario2* ...

Directiva	Comentario:
AuthType	Declara cuál es el tipo de autentificación que va a utilizarse.
AuthName	Indica el nombre de autentificación enviado al programa cliente.
AuthUserFile	Establece el camino para el fichero de usuarios para la autentificación básica (módulo mod_auth).
AuthDigestFile	Establece el camino para el fichero de usuarios para la autentificación MD5 (módulo mod_digest). El administrador del servidor puede usar el programa **htdigest** (situado en el directorio *ServerRoot*/bin) para gestionar este tipo de ficheros. El método MD5 es más seguro que el método básico.
AuthGrupoFile	Indica el camino para el fichero de grupos (módulo mod_auth).

El servidor Apache también soporta la autentificación mediante ficheros DB compatibles con los Unix de Berkeley o mediante usuarios anónimos (equivalentes a los utilizados en el protocolo FTP).

Ejemplo.

En este ejemplo vamos a comentar las características principales para configurar y arrancar un servidor Apache. Asimismo, accederemos a la página que nos confirmará el correcto funcionamiento del programa y verificaremos este hecho revisando los procesos que están ejecutándose en la máquina.

La tabla siguiente presenta el contenido, debidamente comentado, de cada uno de los tres archivos de configuración.

httpd.conf	
#LoadModule mmap_static_module modules/mod_mmap_static.so LoadModule env_module modules/mod_env.so LoadModule config_log_module modules/mod_log_config.so LoadModule agent_log_module modules/mod_log_agent.so LoadModule referer_log_module modules/mod_log_referer.so #LoadModule mime_mhgic_module modules/mod_mime_magic.so LoadModule mime_module modules/mod_mime.so LoadModule negotiation_module modules/mod_negotiation.so LoadModule status_module modules/mod_status.so LoadModule info_module modules/mod_info.so LoadModule includes_module modules/mod_include.so LoadModule autoindex_module modules/mod_autoindex.so LoadModule dir_module modules/mod_dir.so LoadModule cgi_module modules/mod_cgi.so LoadModule asis_module modules/mod_asis.so LoadModule imap_module modules/mod_imap.so LoadModule action_module modules/mod_actions.so #LoadModule speling_module modules/mod_speling.so LoadModule userdir_module modules/mod_userdir.so LoadModule proxy_module modules/libproxy.so LoadModule alias_module modules/mod_alias.so LoadModule rewrite_module modules/mod_rewrite.so LoadModule access_module modules/mod_access.so LoadModule auth_module modules/mod_auth.so LoadModule anon_auth_module modules/mod_auth_anon.so #LoadModule dbm_auth_module modules/mod_auth_dbm.so LoadModule db_auth_module modules/mod_auth_db.so LoadModule digest_module modules/mod_digest.so #LoadModule cern_meta_module modules/mod_cern_meta.so LoadModule expires_module modules/mod_expires.so LoadModule headers_module modules/mod_headers.so LoadModule usertrack_module modules/mod_usertrack.so #LoadModule example_module modules/mod_example.so #LoadModule unique_id_module modules/mod_unique_id.so LoadModule setenvif_module modules/mod_setenvif.so	En primer lugar, se especifican los módulos que serán cargados estática y dinámicamente. La línea que comienza con una almohadilla (#) indica comentario, o sea, dicho módulo no se carga. Por ejemplo, el módulo `mod_mmap_static` se utiliza para conservar en la memoria algunos ficheros, aumentando el rendimiento del servidor. Sin embargo no se carga, por lo tanto, no podrá utilizarse la directiva `MmapFile`. Por otro lado, el administrador ha considerado interesante cargar el módulo `mod_env` –que permite pasar variables de entorno a programas CGI– y sí podrá utilizar las directivas `PassEnv` o `SetEnv`.
ClearModuleList #AddModule mod_mmap_static.c AddModule mod_env.c AddModule mod_log_config.c AddModule mod_log_agent.c AddModule mod_log_referer.c #AddModule mod_mime_magic.c AddModule mod_mime.c AddModule mod_negotiation.c AddModule mod_status.c ...	Apache necesita rehacer la lista de módulos , indicándole el orden correcto de carga.

```
ServerType standalone
Port 80
User nobody
Group nobody
```

Servidor independiente del **inetd**, gestionando el puerto 80.

El supervisor (`root`) es el único usuario que puede iniciar puertos TCP con valor menor a 1.024. Sin embargo –por razones de seguridad– no es conveniente que los procesos del **httpd** sean ejecutados por root. Por ello se definen usuario y grupo ficticios que ejecutan dichos procesos.

```
HostnameLookups off
```

Para mejorar el tiempo de respuesta, no se busca el nombre completo del ordenador cliente, sólo se registra su dirección IP.

```
ServerName betis.musho.es
UseCanonicalName on
ServerAdmin ramon@betis.musho.es
```

Nombre completo del servidor y dirección de correo del administrador del Apache.

```
ServerRoot /etc/httpd
```

Directorio de configuración.

```
ErrorLog logs/error_log
LogLevel warn
```

Localización del fichero de errores (relativa a `ServerRoot`) y nivel de anotaciones.

```
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\"
\"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
```

Formatos de los ficheros históricos de accesos (combinado o normal), de referencias y de clientes.

```
CustomLog logs/access_log common
#CustomLog logs/referer_log referer
#CustomLog logs/agent_log agent
#CustomLog logs/access_log combined
```

En este caso sólo se utiliza un fichero histórico de accesos con formato común.

```
PidFile /var/run/httpd.pid
ScoreBoardFile /var/run/httpd.scoreboard
#LockFile /var/lock/httpd.lock
```

Ficheros especiales (PID del proceso principal, información interna, bloqueos para uso con NFS).

```
Timeout 300
```

Tiempo de espera normal.

```
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
```

Máximo número y tiempo de espera para comunicaciones persistentes.

```
StartServers 10
MinSpareServers 8
MaxSpareServers 20
```

Nº de procesos servidores normales y *de repuesto* usados para atender múltiples peticiones

	para atender múltiples peticiones.
`MaxClients 150`	Nº máximo de clientes que pueden atenderse a la vez.
`MaxRequestsPerChild 100`	Nº máximo de peticiones que puede atender un proceso hijo.
`access.conf`	

`<Directory />`
`Options None`
`AllowOverride None`
`</Directory>`
	Protege todo el árbol de directorio, excepto aquellos que se indiquen en otros grupo `</Directory>`.

`<Directory /home/httpd/html>`	Directorio púbico (indicado por la directiva `DocumentRoot`).
`Options Indexes Includes FollowSymLinks` `AllowOverride None`	Opciones aplicables al árbol de directorios del servidor. No podrán ser modificadas por ficheros `.htaccess`.

`order allow,deny`
`allow from all`
`</Directory>`
	Acceso global a esta información.

`<Directory /home/httpd/cgi-bin>`
`AllowOverride None`
`Options ExecCGI`
`</Directory>`
	Directorio de programas (indicado por `ScriptAlias`).

`<Location /server-status>`
`SetHandler server-status`
`order deny,allow`
`deny from all`
`allow from .musho.es`
`</Location>`
	URL que muestra un informe sobre el estado del servidor, con acceso único desde las máquinas del dominio `.musho.es`.

`<Location /cgi-bin/phf*>`
`deny from all`
`ErrorDocument 403 /cgi-bin/error_phf.cgi`
`</Location>`
	Registra los intentos de acceso al servidor a través de un fallo en las primeras versiones de Apache. El gestor debe crear el programa **`error_phf.cgi`**, que registre los datos del posible agresor.

`srm.conf`	
`DocumentRoot /home/httpd/html`	Directorio raíz de los documentos del servidor.
`UserDir public_html`	Directorio raíz de los documentos de cada usuario (relativo a su `$HOME`).
`DirectoryIndex index.html index.shtml index.cgi`	Índice HTML de un directorio.
`FancyIndexing on`	Genera índices con los archivos

de un directorio si no existe ninguno de los archivos indicados en `DirectoryIndex`.

```
AddIconByEncoding (CMP,/icons/compressed.gif) x-compress
x-gzip
AddIconByType (TXT,/icons/text.gif) text/*
AddIconByType (IMG,/icons/image2.gif) image/*
AddIconByType (SND,/icons/sound2.gif) audio/*
AddIconByType (VID,/icons/movie.gif) video/*
AddIcon /icons/binary.gif .bin .exe
AddIcon /icons/binhex.gif .hqx
AddIcon /icons/tar.gif .tar
AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
AddIcon /icons/a.gif .ps .ai .eps
...
AddIcon /icons/back.gif ..
AddIcon /icons/hand.right.gif README
AddIcon /icons/folder.gif ^^DIRECTORY^^
AddIcon /icons/blank.gif ^^BLANKICON^^
DefaultIcon /icons/unknown.gif
```

Cuando el índice de un directorio se genera automáticamente, estas directivas permiten asociar iconos a grupos de ficheros, según su codificación, su tipo o su extensión.

```
ReadmeName README
HeaderName HEADER
```

Archivos descriptivo y de cabecera del directorio.

```
IndexIgnore .??* *~ *# HEADER* README* RCS .htaccess
```

Archivos que se ignoran al generar el índice. Es recomendable añadir una referencia al fichero `.htaccess`.

```
AccessFileName .htaccess
```

Archivo modificador de las opciones de acceso al directorio.

```
TypesConfig /etc/mime.types
AddEncoding x-compress Z
AddEncoding x-gzip gz
DefaultType text/plain
```

Codificación de los grupos de ficheros y tipo de fichero por omisión.

```
AddLanguage es .es
AddLanguage en .en
AddLanguage fr .fr
AddLanguage de .de
LanguagePriority es en fr de
```

Codificación y prioridad de sufijos de nombres de archivos para indicar el idioma de cada documento.

```
Redirect permanent /datos http://nuevo.servidor.es/datos/
Redirect temp      /pruebas /DirTemporal
```

Establece los nuevos URLs de documentos que han sido movidos de lugar.

```
Alias /icons/ /home/httpd/icons/
ScriptAlias /cgi-bin/ /home/httpd/cgi-bin/
```

Alias para directorios específicos de iconos y de programas.

```
AddHandler cgi-script .cgi
AddType text/html .shtml
AddHandler server-parsed .shtml
AddHandler imap-file map
```

Asociación de tipos de ficheros con acciones específicas de Apache (ficheros HTML o mapas analizados por el servidor, programas CGI, etc.).

```
#ErrorDocument 404 "Dcomento no encontrado.
#ErrorDocument 404 /error/NoHallado.html
ErrorDocument 404 /cgi-bin/error.pl
```

Personalizar las respuestas de errores del servidor mediante textos, documentos HTML fijos o programas CGI.

```
BrowserMatch "Mozilla/2" nokeepalive
BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0
force-response-1.0
BrowserMatch "RealPlayer 4\.0" force-response-1.0
BrowserMatch "Java/1\.0" force-response-1.0
BrowserMatch "JDK/1\.0" force-response-1.0
```

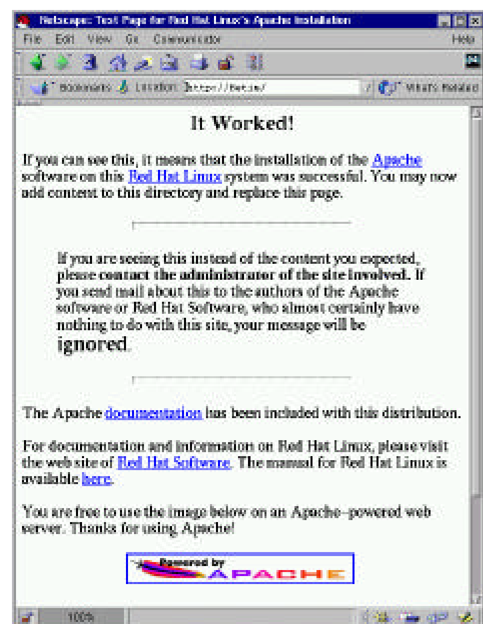
Estas directivas corrigen algunos errores de comunicación con ciertas versiones de algunos programas clientes.

Una vez realizada la configuración completa del servidor, éste debe arrancarse, utilizando el guión correspondiente para la gestión de servicios. En nuestro caso, ejecutar:

```
# /etc/rc.d/init.d/httpd start
```

Cada vez que el administrador deba realizar cambios en la configuración del servidor tendrá que pararlo y volverlo a arrancar.

Entonces, ya pueden crearse los programas de mantenimiento y las páginas de información, que serán visualizadas con cualquier programa *navegador*.



Comprobación del Apache

La figura contigua muestra la página de comprobación del servidor Apache para Linux de RedHat. Para ello se ha ejecutado el programa Netscape Communicator (tecleando desde el terminal X: **`netscape &`**) e indicando al *naveador* la dirección principal del servidor (`http://Betis/`).

Nótese que no hemos indicado la dirección completa del servidor, ya que tanto el cliente como el servidor están siendo ejecutados en ordenadores del mismo dominio.

Para comprobar el efecto real sobre la máquina, podemos observar los procesos del Apache que están siendo ejecutados en el ordenador.

```
# ps axu | grep httpd
nobody     301  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     302  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     303  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     304  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     305  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     306  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     307  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     309  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     310  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
nobody     311  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
root       298  0.0  1.7  1916  1132  ?  S   18:25   0:00 httpd
```

Servidor para la Transferencia de Ficheros (FTP)

El servicio de transferencia de ficheros se utiliza para enviar o recibir archivos de un ordenador a otro. El **FTP** (*File Transfer Protocol*) es uno de los protocolos más antiguos utilizados en Internet, sin embargo sigue siendo ampliamente utilizado.

En un principio, el usuario de un programa cliente FTP tenía que escribir los mandatos del protocolo para realizar una transferencia (el mandato **`ftp`** incluido en la mayoría de los Unix trabaja de esta manera). Hoy día existen multitud de clientes gráficos, que permiten trabajar más cómodamente; incluso los *navegadores* suelen incluir la posibilidad de recibir información mediante el protocolo FTP. También hay programas que permiten trarse *en diferido* –sin la intervención del usuario– una copia de ciertas partes del árbol de directorios públicos de un servidor.

Los programas servidores de FTP también han sufrido una cierta evolución, incluyendo características como el acceso de usuarios anónimos o la conversión del formato de archivos de texto entre Unix y MS-DOS.

La transferencia de ficheros se basa en:

- Un protocolo de transporte conocido como FTP, que utiliza dos puertos TCP. La norma indica que el puerto 20/TCP se utiliza para el intercambio de órdenes, mientras que el 21/TCP, para la transferencia de datos. Ambas líneas deben incluirse en el fichero `/etc/services`.
- Una serie de mandatos conocidos por los programas que establecen la comunicación. El usuario que utilice clientes FTP en modo texto deberá estar familiarizado con las primitivas del protocolo.
- Una estructura cliente/servidor, donde la información de los mandatos es enviada por el programa cliente y procesada por servidor.
- El servidor puede permitir el acceso a los usuarios reales del sistema –los cuales podrán operar libremente con sus datos– o a usuarios anónimos –que suelen tener un acceso más restringido.

Un usuario anónimo es aquél que no tiene una cuenta real en el ordenador, se identifica por la palabra `anonymous` y utiliza como clave la dirección de correo electrónico de la persona. Este usuario sólo tiene acceso al árbol de directorios público del sistema y en ningún caso *verá* o accederá al resto de información del sistema. No es recomendable que los usuarios anónimos puedan escribir o modificar los datos públicos del servidor.

Configuración previa.

Antes de entrar a describir y a configurar el programa servidor FTP, el administrador del sistema deberá realizar algunas comprobaciones previas.

- Revisar el fichero `/etc/services` y comprobar que existen las dos líneas correspondientes a los servicios del FTP.

```
ftp-data  20/tcp
ftp       21/tcp
```

- Comprobar o añadir la línea correspondiente al servidor FTP en el archivo de configuración del **`inetd`**.

```
ftp  stream  tcp  nowait  root  Servidor  Argumentos
```

- Si el servidor va a permitir el acceso anónimo, deberá crearse un usuario ficticio `ftp` con las siguientes características:
  - No tendrá posibilidad de acceder normalmente a la máquina (no tendrá clave).
  - Pertenecerá al grupo `ftp`.
  - El directorio de su cuenta corresponderá con la raíz del árbol público.

El servidor WU-FTPd.

El **WU-FTPd** –creado por la Universidad de Washington– es uno de los servidores para la transferencia de ficheros de dominio público más utilizados en Internet y resulta muy fácil de configurar. Sus características principales son:

- Soporta la conexión como usuario real (si está definido en el fichero de claves del sistema) o como usuario anónimo.
- Pueden definirse varias clases de usuarios, dependiendo de la dirección IP o del nombre de la máquina cliente y del tipo de usuario conectado.
- Establece límites de conexión y restricciones de acceso para cada clase de usuarios.
- Pirmite el uso de ficheros históricos para registrar los accesos al servidor.
- Incluye las siguientes características operativas para los usuarios anónimos:
  - Directorio público utilizado como almacén de ficheros.
  - El usuario público sólo puede ver el árbol público.
  - Puede crearse un directorio para añadir archivos (es poco recomendable).

- Utilización de varios tipos de mandatos (borrar, modificar, cambiar permisos, etc.).
- Incluir mensajes de bienvenida e índices para cada directorio.

El árbol público deberá contar, al menos, con los siguientes directorios:

`bin`:	Incluye los mandatos necesarios para listar directorios, comprimir o descomprimir archivos y cualquier otra operación que el gestor del FTP permita realizar al usuario anónimo.
`lib`:	Deben añadirse las bibliotecas del sistema operativo necesarias para ejecutar los mandatos del directorio `bin`.
`etc`:	Contiene los archivos de usuarios y grupos propietarios de los archivos públicos. **No incluir claves de acceso en estos ficheros.**
`pub`:	En él se almacenan todos los ficheros y directorios públicos.

Configuración del WU-FTPd.

El servidor FTP de la Universidad de Washington se configura editando una serie de ficheros que almacenan dicha información. Si el usuario administrador considera oportuno modificarla mientras el programa está activo, deberá reiniciar el **`inetd`** después de reconfigurar el WU-FTPd.

En primer lugar, veremos los distintos tipos de usuarios que pueden definirse en este servidor:

- Real: todo usuario propio de la máquina (con una entrada en el archivo `/etc/passwd`).
- Anónimo: usuario ficticio que cumple las siguientes características:
    - Utilizará como nombre de usuario la palabra `anonymous`.
    - La clave coincidirá con la dirección de correo electrónico de la persona anónima
    - Sólo podrá acceder al árbol de directorios reservado para el acceso anónimo.
    - No podrá ejecutar mandatos `USER` ni `PASS` –propios del protocolo FTP– para cambiar su nombre de usuario o su clave sin salir antes de la sesión.
    - El nombre de usuario real para el acceso anónimo es `ftp`.
- Invitado: todo usuario real perteneciente al grupo invitado (directiva `guestgroup`) accederá a los datos de su cuenta como usuario anónimo, por lo tanto no podrá *ver* el resto del árbol de directorios, ni podrá modificar su nombre de usuario ni su clave sin salir antes de la sesión.

Resulta importante hacer notar que los archivos de configuración del WU-FTPd se encuentran normalmente situados en el directorio `/etc`, aunque no es una ubicación obligatoria. El administrador puede compilar una versión nueva del programa y definir un directorio de configuración diferente.

El fichero principal para configurar el servidor es `/etc/ftpaccess`, que consta de los 5 grupos de directivas descritos en la próxima tabla.

Directivas de acceso:	
`class`	Define clases de usuarios dependiendo de su tipo (real, invitado o anónimo) y de la localización del ordenador cliente utilizado.
`deny`	Niega el acceso a determinados ordenadores.
`autogroup`	Permite que los usuarios anónimos de una determinada clase obtengan permiso efectivo para un determinado grupo de usuarios.
`guestgroup`	Todo usuario perteneciente a los grupos invitados se conecta a su cuenta como anónimo (no puede acceder a otras zonas del árbol de directorios).
`limit`	Limita el nº de usuarios de una determinada clase que pueden acceder a la vez al servidor.
`loginfails`	Nº de fallos de conexión permitidos.
`private`	Refuerza el acceso privado al servidor, utilizando un fichero de acceso para grupos de usuarios (`/etc/ftpgroup`).
`noretrieve`	Niega el acceso a determinados ficheros.
Directivas informativas:	
`banner`	Muestra un mensaje de presentación antes de pedir usuario y clave.
`message`	Muestra el contenido de un archivo cuando el usuario se conecta o cuando entra en un nuevo directorio. Este fichero puede incluir caracteres de control que dan información adicional, (hora local, espacio libre, directorio actual, nº de usuarios conectados, etc.).
`readme`	Notificará a ciertos usuarios -cuando se conecten o accedan a algún directorio- las fechas de última modificación de determinados ficheros.
`email`	Dirección de correo electrónico del administrador del servidor.
Directivas para registro histórico:	
`log commands`	Guarda un histórico de los mandatos ejecutados por cada tipo de usuarios.
`log transfers`	Alamacena un histórico de las transferencias realizadas por cada tipo de usuarios.
Directivas de permisos:	
`passwd-check`	Tipo de comprobación de claves para usuarios anónimos.

`path-filter`	Indica los ficheros que no se incluirán en expresiones regulares.
`upload`	Establece las características de aquellos directorios donde se permite exportar archivos hacia el servidor.

`chmod`
`delete`
`overwrite`
`rename`
`umask`
`compress`
`tar`
	Estas directivas representan las acciones o mandatos que pueden ser ejecutados o denegados para cada tipo de usuarios (real, invitado o anónimo). Por omisión, todos están activas. Los usuarios anónimos sólo deben tener permitda la ejecución de mandatos **`compress`** y **`tar`**.

Directivas varias:	
`shutdown`	Establece el camino para el fichero que indica la fecha y la hora para una caída planificada del servidor FTP. Dicho fichero puede generarse utilizando el mandato **`ftpshut`**.
`alias`	Define alias para directorios.
`cdpach`	Incluye caminos de búsqueda para subdirectorios.
`virtual`	Define las características operativas de un servidor FTP virtual.

La siguiente tabla describe el resto de los ficheros de configuración del WU-FTPd.

`/etc/ftpusers`	
Descripción:	Contiene la lista de usuarios que no podrán utilizar el servidor FTP. **Deben añadirse a esta lista todos los usuarios ficticios y de administración de la máquina.**
Formato:	*`Usuario`* `...`

Ejemplo:	`root`
	`bin`
	`daemon`
	`adm`
	`lp`

`/etc/ftpgroups`	
Descripción:	Formato del archivo para acceso a grupos (ver directiva `private`).
Formato:	*`GrupoAcceso`*`:`*`ClaveEncriptada`*`:`*`GrupoReal`* `...`
Ejemplo:	`invitado:dfF.3Ak9+mQ2U:games`
`/etc/ftphosts`	
Descripción:	Permite o niega el acceso a ciertas cuentas desde las máquinas especificadas.

Formato:	**allow** *Usuario Dirección* [*Dirección* ...] ... **deny** *Usuario* **Dirección** [*Dirección* ...] ... **Nota:** El usuario `ftp` indica usuario de tipo anónimo.
Ejemplo:	`allow ramon *.facultad.us.es 150.214.*` `deny  admin *.piratas.com`
/etc/ftpconversions	
Descripción:	Contiene la base de datos para la conversión de tipos de archivos.
Formato:	:*Extensión*: :*Postfijo*:*Mandato*:*Tipos*:*Opciones*:*Descrip* ...
Ejemplo:	`:   : :.gz:/bin/gzip -9 -c %s:T_REG:O_COMPRESS:GZIP` `:   : :.tar:/bin/tar -c -f - %s:T_REG|T_DIR:O_TAR:TAR`

WU-FTPd incluye una serie de mandatos para monitorizar y administrar el acceso al servidor. La siguiente tabla describe cada uno de ellos.

`ftpwho`:	Lista todos los usuarios que están conectados al servidor FTP.
`ftpcount`:	Indica cuántos usuarios de cada clase hay conectados.
`ftpshut`:	Establece los parámetros para una parada programada del servidor.

Ejemplo.

En primer lugar se verán los archivos típicos para configurar el WU-FTPd; posteriormente se incluirán ejemplos para los distintos ficheros informativos, que permiten mejorar el acceso anónimo; y en último lugar se observarán las salidas de los mandatos de administración incluidos en el paquete.

ftpaccess	
`banner /etc/ftp/caratula.ftp`	Indica el archivo con la cabecera de entrada al servidor.
`loginfails 2`	Nº de intentos de acceso.
`class local real,guest,anonymous *.facultad.us.es 0.0.0.0` `class remoto real,guest,anonymous *`	Define 2 clases de usuarios: una para los que acceden desde máquinas de la empresa y otra para el acceso remoto.
`guestgroup otros`	Los usuarios del grupo `otros` acceden como invitados.

`limit local 60 Any /etc/ftp/mucho.local`
`limit remoto 100 SaSu|Any1800-0600 /etc/ftp/mucho.remoto`
`limit remoto 50 Any /etc/ftp/mucho.remoto`
	Máximo nº de usuarios por clase. Imprime un mensaje si se sobrepasa el límite establecido.

`message /.bienvenida login`
`message .mensaje cwd=*`
`readme README* login`
`readme README* cwd=*`
	Ficheros informativos.

`compress yes local remoto`
`tar yes local remoto`
`delete no anonymous,guest`
`overwrite no anonymous,guest`
`rename no anonymous,guest`
`chmod no anonymous`
`umask no anonymous`
	Mandatos permitidos y denegados para cada clase y cada tipo de usuarios.

`private yes`	Permite usar archivos de acceso privado.
`passwd-check rfc822 warn`	Pide una dirección de correo electrónico como clave anónima.
`log commands real` `log transfers anonymous,real inbound,outbound`	Registra todas las transferencias, pero sólo los mandatos ejecutados por usuarios reales.
`shutdown /etc/ftp/caida.ftp`	Fichero con los datos de la próxima parada programada.

`upload /home/ftp * no`
`upload /home/ftp /incoming yes root adm 0640 dirs`
`upload /home/ftp /bin no`
`upload /home/ftp /etc no`
	Permite crear archivos y directorios públicos en el directorio `/home/ftp/incoming`.

`alias practicas /pub/doc/practicas`	Abreviaturas (alias).
`cdpath /pub` `cdpath /mirror`	Camino de búsqueda para cambiar de directorio.
`path-filter anonymous ^[-A-Za-z0-9_\.]*$ ^\. ^-` `path-filter guest ^[-A-Za-z0-9_\.]*$ ^\. ^-`	Los usuarios anónimos e invitados no podrán ver los archivos que comiencen por el carácter ^.
`email ramon.gomez@fie.us.es`	Dirección de correo electrónico del administrador del servidor FTP.
ftpusers	

`root`
`bin`
`daemon`
`adm`
`lp`
`sync`
`shutdown`
`halt`
`mail`
`news`
	Lista de usuarios que serán rechazados al intentar conectarse al servidor FTP. Se incluyen todos los usuarios administradores y ficticios del Linux de RedHat 5.2 (excepto `ftp`, claro).

<table>
<tr><td>uucp
operator
games
nobody</td><td></td></tr>
<tr><th colspan="2">ftpconversions</th></tr>
<tr><td><pre> :.Z: : :/bin/compress -d -c
%s:T_REG|T_ASCII:O_UNCOMPRESS:UNCOMPRESS
 :   : :.Z:/bin/compress -c %s:T_REG:O_COMPRESS:COMPRESS
 :.gz: : :/bin/gzip -cd
%s:T_REG|T_ASCII:O_UNCOMPRESS:GUNZIP
 :   : :.gz:/bin/gzip -9 -c %s:T_REG:O_COMPRESS:GZIP
 :   : :.tar:/bin/tar -c -f - %s:T_REG|T_DIR:O_TAR:TAR
 :   : :.tar.Z:/bin/tar -c -Z -f -
%s:T_REG|T_DIR:O_COMPRESS|O_TAR:TAR+COMPRESS
 :   : :.tar.gz:/bin/tar -c -z -f -
%s:T_REG|T_DIR:O_COMPRESS|O_TAR:TAR+GZIP</pre></td><td>Base de datos para la conversión de tipos de ficheros.

Establece reglas para que el servidor se encargue de las operaciones de compresión, descompresión, archivado o extracción de datos.</td></tr>
</table>

<table>
<tr><th colspan="2">/etc/ftp/caratula.ftp</th></tr>
<tr><td><pre>   FTP.FACULTAD.US.ES

Servidor FTP anónimo de mi Facultad
      Universidad de Sevilla</pre></td><td>Mensaje de la carátula de entrada al servidor FTP, que se muestra antes de pedir usuario y clave (ver directiva <code>banner</code>).</td></tr>
<tr><th colspan="2">/etc/ftp/mucho.local</th></tr>
<tr><td><pre>Ahora hay demasiados alumnos conectados al servidor de la
Facultad.
Inténtalo dentro de un rato.</pre></td><td>Se muestra este mensaje si se sobrepasa el tope de usuarios para la clase local (ver directiva <code>limit</code>).</td></tr>
<tr><th colspan="2">/etc/ftp/mucho.remoto</th></tr>
<tr><td><pre>Saturación del servidor.
Hay muchos usuarios conectados.
Pruebe a conectarse dentro de unos instantes.</pre></td><td>Se muestra este mensaje si se sobrepasa el tope de usuarios para la clase remoto (ver directiva <code>limit</code>).</td></tr>
<tr><th colspan="2">/etc/ftp/caida.ftp</th></tr>
<tr><td><pre>1999 05 15 13 30 0100 0010
Caida del servidor FTP prevista para %s.</pre></td><td>Caída programada a las 13:30 del 15 de <u>junio</u> de 1.999 (atención al formato de la fecha). No se aceptan peticiones desde una hora antes de la caída y las conexiones se cortan 10 minutos antes. Se imprime un mensaje indicando la hora de la parada.</td></tr>
</table>

El usuario `ftp` se utiliza para el acceso anónimo al servidor. Teniendo en cuenta que su directorio raíz es el `/home/ftp`, veremos algunos ejemplos de archivos informativos.

<table>
<tr><th colspan="2"><code>/home/ftp/.bienvenida</code></th></tr>
<tr><td><code>El usuario con e-mail %u y conectado desde %R,
ha accedido a la cuenta $U del servidor %L.
La hora local es %T.
Hay %N usuarios de un máximo de %M.
Para cualquier consulta, mandar correo a %E.</code></td><td>Mensaje de bienvenida que se muestra al usuario después de validar su nombre y su clave (ver directiva <code>message login</code>).</td></tr>
<tr><th colspan="2"><code>/home/ftp/pub/.mensaje</code></th></tr>
<tr><td><code>Directorio actual: %C

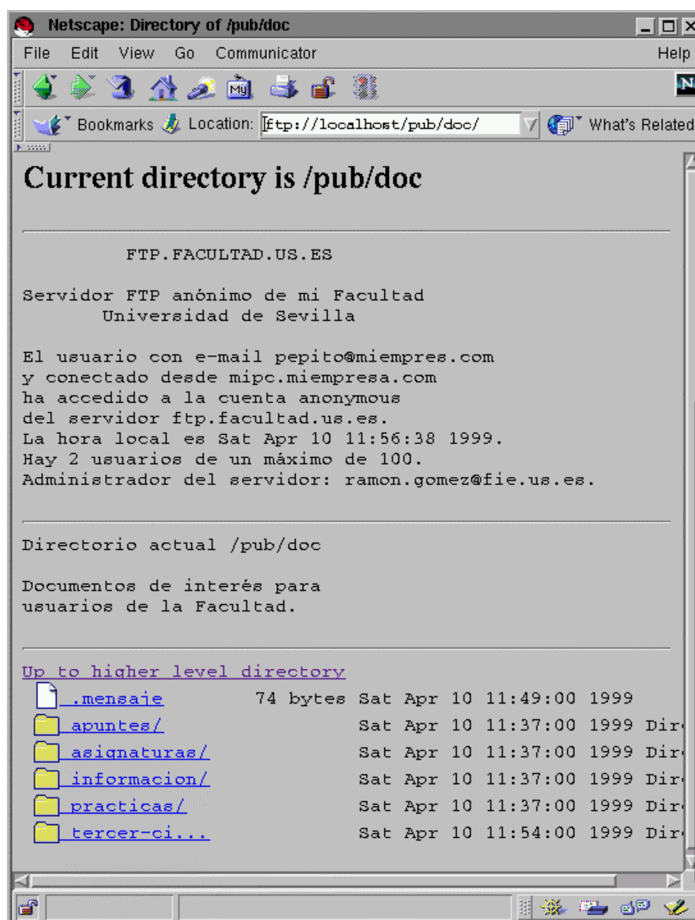
Este directorio contiene los
ficheros públicos del servidor.</code></td><td>Mensaje que se presenta cuando el usuario entra por primera vez en el directorio <code>/pub</code> del árbol anónimo (ver directiva <code>message cwd</code>).</td></tr>
<tr><th colspan="2"><code>/home/ftp/pub/doc/.mensaje</code></th></tr>
<tr><td><code>Directorio actual: %C

Documentos de interés para
usuarios de la Facultad.</code></td><td>Mensaje para el directorio <code>/pub/doc</code> del árbol anónimo (ver directiva <code>message cwd</code>).</td></tr>
</table>

La siguiente imagen describo el uso del *navegador* Netscape Communicator para acceder al servidor FTP del ejemplo. Para obtener una información completa, el usuario ha tecleado la siguiente dirección:

```
ftp://ftp.facultad.us.es/pub/doc
```

El servidor muestra la carátula de entrada (es la primera conexión al servidor), el mensaje del directorio público `/pub/doc` y el contenido de dicho directorio.



Acceso al FTP anónimo.

Servidor de correo electrónico.

El correo electrónico (conocido también por la abreviatura en inglés: *e-mail*) es otro de los servicios principales y más utilizados en Internet. Tiene una gran similitud con el correo normal, ya que permite enviar mensajes al buzón de un usuario y éste podrá recogerlo cuando estime oportuno.

Al igual que en los casos anteriores, este servicio también a evolucionado en gran medida. Los primeros programas para el tratamiento del correo electrónico sólo permitían enviar y recibir mensajes de texto. Actualmente, los clientes pueden anexar todo tipo de ficheros, imágenes o datos a los mensajes de correo. Los programas *navegadores* más importantes incluyen funciones para el tratamiento y la recepción de correo electrónico, soportando que los propios mensajes puedan tener el formato de páginas de hipertexto y hacer referncia a todo tipo de recursos publicados en la red.

También ha variado la manera de recoger los mensajes almacenados en el buzón. Antes el usuario debía conectarse a una cuenta en el ordenador servidor de correo y ejecutar el programa –normalmente en modo texto– que le permitía leer, almacenar, borrar o enviar mensajes (**`mail`** o **`mailx`**, PINE, ELM).

Hoy día puede leerse el correo desde un ordenador personal utilizando una clave de acceso personal al servidor –sin necesidad de conectarse a él vía Telnet– mediante un cliente gráfico o un *navegador*, puede importarse el correo al propio PC o dejarlo almacenadado en el servidor para una revisión posterior, pueden leerse los mensajes desde distintas máquinas o clasificarse en distintas carpetas.

La Dirección de cada buzón de correo electrónco –igual que ocurre con la de un buzón normal– debe ser única en toda la red, sin embargo existe la posibilidad de definir alias o sobrenombres que identifiquen al mismo buzón.

Hay dos formas para identificar una dirección de correo electrónico:

- Indicando el nombre de usuario del destinatario (o su alias de correo) y la dirección completa de la máquina servidora.

  `Buzón@DirecciónDelOrdenador`

- Señalar el alias de correo para el destinatario –o su nombre de usuario en el servidor– y el dominio correspondiente a la empresa u organización donde la persona recoge sus mensajes. Esta opción debe indicarse expresamente en la configuración del servidor de correo electrónico.

  `Buzón@DominioDeRed`

Los alias de correo se usan para establecer identificadores más comprensibles para los nombres de usuarios definidos en la máquina o para añadir listas de correo. Una **lista de distribución de correo** es un alias que está asociado a varios buzones, por lo tanto cuando se envía un mensaje a

la lista, éste se dirige a los buzones de todos sus miembros.

El siguiente ejemplo muestra la dirección del buzón de correo del ponente de este curso.

Servidor:	`antena.fie.us.es`
Usuario:	`ramon`
Alias:	`ramon.gomez`
Dirección de correo:	`ramon.gomez@fie.us.es`

Otras direcciones para el mismo buzón:	`ramon@fie.us.es`
	`ramon@antena.fie.us.es`
	`ramon.gomez@antena.fie.us.es`

Los modernos programas de correo electrónico soportan la definición de tarjetas de correo, que asocian con la dirección de correo electrónico otros datos útilies de interés sobre el destinatario (nombre, empresa, dirección real, etc.).

Otro avance importante es el protocolo **MIME**, que permite definir tipos de ficheros basados en sus extensiones, que pueden ser incluidos como anexos de los mensajes. El destinatario deberá configurar sus aplicaciones para leer cada tipo de fichero.

El servidor Sendmail.

Sendmail –creado en la Universidad de Berkeley (EE.UU.)– es el proceso que controla la transferencia de mensajes de correo electrónico. Dicha transferencia sigue los siguientes pasos:

1. El usuario emisor utiliza un programa de correo para escribir su mensaje.
2. El cliente de correo envía su mensaje al Sendmail para que lo traduzca al protocolo **SMTP** (*Simple Mail Transfer Protocol*).
3. El Sendmail del ordenador receptor escucha permanentemente el puerto 25/TCP y –al detectar una conexión– procesa la información, convierte el mensaje al formato adecuado y lo almacena en el buzón correspondiente.
4. El usuario receptor lee su buzón utilizando un cliente de correo.

Como veremos en el siguiente apartado, no es necesario que todos los ordenadores tengan activado el proceso Sendmail, ya que hay programas de correo electrónico que permiten enviar y recibir correo de servidores remotos. Sin embargo, los ordenadores servidores de correo sí utilizan Sendmail para transferirse los mensajes.

Las características principales de Sendmail son:

- Es un proceso que se ejecuta independientemente del **`inetd`**.

- El fichero principal de configuración –`/sendmail.cf`, situadio normalmente en `/etc` o en `/etc/mail`– consta de una serie de reglas bastante complejas.
- Pueden definirse alias o apodos para los usuarios de correo electrónico, almacenados en el fichero `/etc/aliases`. Este archivo permite definir también listas de correo.
- Los ficheros `/etc/mime.types` y `/etc/mailcap` definen los anexos multimedia que pueden incluirse en los mensajes de correo y las aplicaciones que los gestionan.

Configuración del Sendmail.

El archivo de configuración `sendmail.cf` consta de una gran cantidad de reglas bastante engorrosas y complejas que no serán tratadas en este curso. Sin embargo **RedIRIS ofrece un servicio, a través de hipertexto, para la generación rápida y sencilla de este tipo de archivos, en la dirección** http://www.rediris.es/mail/generador/sendmail-30.es.html.

Como se muestra en las figuras adjuntas, el administrador sólo deberá introducir los datos básicos de su sistema y el programa devolverá el archivo de configuración adecuado.

Los datos que deben ser introducidos son:

- Sistema operativo.
- Nivel de la estafeta de correos, que debe ser acordado entre el gestor del servidor de correo y el administrador del servidor de nombres. Estos niveles son:
  1. Estafeta principal del dominio.
  2. Intercambia con la estafeta de nivel 1 todo el correo interno al dominio.
  3. Dirige todo el correo a la estafeta

Configuración automática de Sendmail en RedIRIS (1/2).

de nivel 1.

- Dominio de correo.
- Indicar si las direcciones usan el dominio o el nombre completo de la máquina.
- Establecer los posibles dominios virtuales gestionados por la estafeta.
- Incluir las máquinas marcadas en el servicio de nombre como MX (intercambio de correo). Suele estar desactivado.
- Permitir soporte para alias inversos.
- Establecer mecanismos de seguridad. Suelen utilizarse los opciones predefinidas.

Configuración automática de Sendmail en RedIRIS (2/2).

Después de rellenar este formulario, el servicio de RedIRIS envía un archivo de configuración válido. La siguiente tabla muestra únicamente la sección de información local (“*local info*”) del fichero `sendmail.cf` obtenido para el ejemplo.

```
# level 8 config file format
V8/Berkeley

##################
#   local info   #
##################

Cwlocalhost

# my official domain name
# ... define this only if sendmail cannot automatically determine your
domain
#Dj$w.Foo.COM

CP.

#  BITNET relay host
DBsmtp:gumncc.terena.nl
CPBITNET

# "Smart" relay host (may be null)
DS
```

```
# operators that cannot be in local usernames (i.e., network indicators)
CO @ % !

# a class with just dot (for identifying canonical names)
C..

# a class with just a left bracket (for identifying domain literals)
C[[

# Mailer table (overriding domains)
Kmailertable hash -o /etc/mailertable

# Resolve map (to check if a host exists in check_mail)
Kresolve host -a<OK> -T<TEMP>

# Hosts that will permit relaying ($=R)
FR-o /etc/mail/relay-domains

# who I send unqualified names to (null means deliver locally)
DR

# who gets all local email traffic ($R has precedence for unqualified
names)
DH

# dequoting map
Kdequote dequote

# class E: names that should be exposed as from this host, even if we
masquerade
# class L: names that should be delivered locally, even if we have a
relay
# class M: domains that should be converted to $M
#CL root
CE root

# who I masquerade as (null for no masquerading) (see also $=M)
DMfacultad.us.es

# my name for error messages
DnMAILER-DAEMON

# Mail to this domain will be considered internal
D{InDom}facultad.us.es

# Aliasrev map definition
Kaliasrev btree -o /etc/aliasrev

# Configuration version number
DZ8.9.1
```

Para más información sobre el correo electrónico y sobre el programa Sendmail, revisar las páginas de manual correspondientes al mandato **`sendmail`** o buscar información en la direción http://www.sendmail.org/.

El servidor POP3.

Hay varios protocolos que permiten enviar y recoger los mensajes al igual que una persona utiliza una oficina de correos. Uno de los más usados es el **POP3** (*Post Office Protocol version 3*).

Los pasos que sigue un mensaje de correo son:

1. El emisor escribe su mensaje en un programa configurado para trabajar con POP3.
2. El programa envía los datos al servicor POP3.
3. El programa servidor POP3 recoge el mensaje y lo envía al Sendmail.
4. El Sendmail del servidor remitente envía el mensaje al servidor destinatario.
5. El Sendmail del servidor receptor almacena el mensaje en el buzón correspondiente.
6. El usuario receptor utiliza un cliente de correo configurado para usar su servidor como estafeta POP3.
7. El usuario debe autentificarse en su estafeta.
8. El servidor POP3 indica al cliente de correo que tiene mensajes nuevos y se los envía.
9. El usuario lee sus mensajes.

Configuración del POP3.

El proceso que gestiona la estafeta de correos electrónica se ejecuta a través del **`inetd`**, por lo tanto deberá existir una línea correspondiente en el fichero `/etc/services` y otra en el `/etc/inetd.conf`.

```
$ grep pop-3 /etc/services
pop-3      110/tcp
$ grep pop-3 /etc/inetd.conf
pop-3  stream tcp nowait root /usr/sbin/tcpd  ipop3d
```

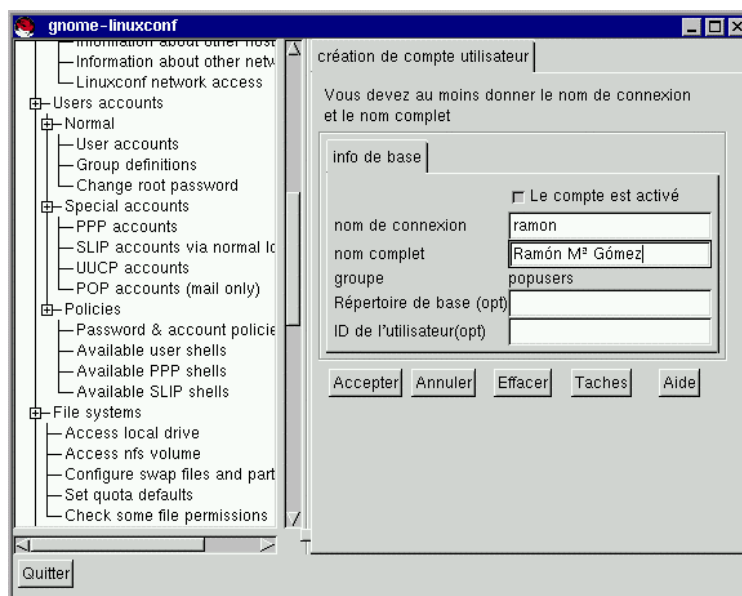
Como se observa en la línea de `inetd.conf`, el proceso **`ipop3d`** ve reforzada su seguridad con el filtro TCP-Wrappers (**`tcpd`**) que permite restringir las peticiones de servicio sólo a los ordenadores autorizados.

El administrador del servidor está encargado de crear las cuentas que permiten al usuario recoger sus mensajes de la estafeta.

La siguente figura muestra el uso de **Linuxconf** para crear fácilmente las cuentas POP. El programa solicita los siguientes datos:

- Activar o no el indicador de cuenta disponible.

- Nombre de usuario.
- Nombre completo de la persona.
- Opcionalmente, directorio de la cuenta. El valor por omisión es `/home/`***`Usuario`***.
- Opcionalmente, identificador de usuario. Si no se establece ningún valor, el sistema lo asignará automáticamente.
- Si la cuenta es activa, al administrador deberá asignar una clave de acceso.



Creación de cuentas POP con Linuxconf.

Seguridad.

El correo electrónico es uno de las utilidades que deben requerir la atención del administrador, ya que puede convertirse en puerta de entrada al sistema y en medio de propagación de virus informáticos. Por lo tanto, algunas de las acciones de seguridad más importantes corresponden también a los usuarios de este servicio.

Las recomendaciones de seguridad más importantes son:

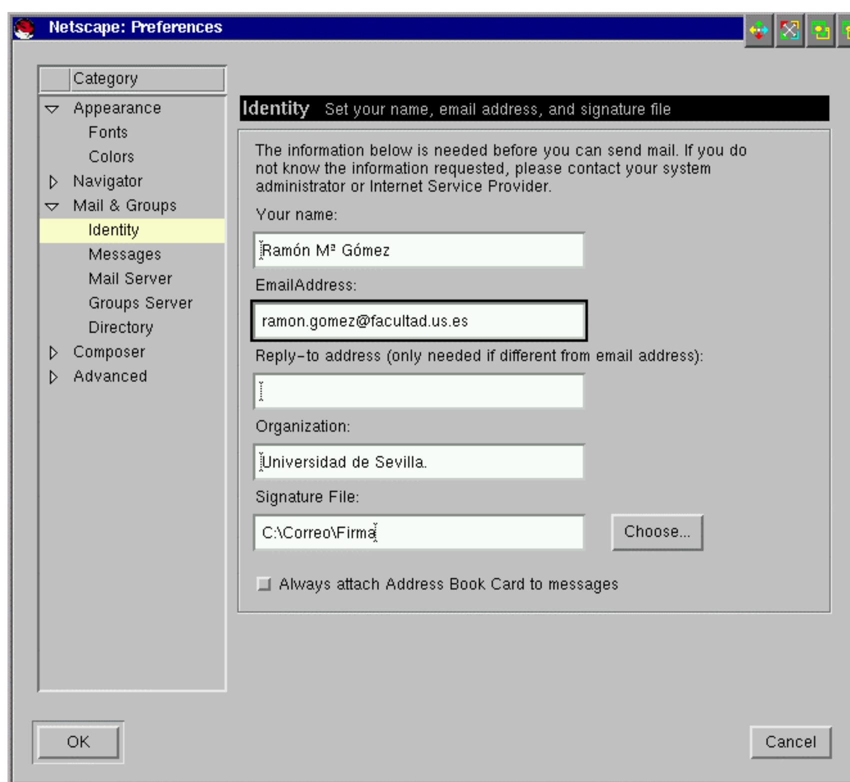
- El usuario administrador deberá utilizar una versión segura de Sendmail para el sistema operativo del servidor. En algunos casos puede ser necesario actualizar la versión del Sendmail.
- Almacenar los buzones de correo en un sistema de archivos independiente, para evitar que ciertos mensajes saturen el disco duro y afecten al trabajo de los usuarios de la máquina. El sistema de archivos `/var` suele usarse para guardar todo tipo de información variada.
- Si se utilizan protocolos para la recogida de mensajes –como el POP3–, activar únicamente Sendmail en los ordenadores servidores de correo.
- Recomendar a los usuarios que revisen y eliminen periódicamente sus mensajes, para recuperar espacio de disco no utilizado. El administrador puede crear programas para vaciar aquellos buzones que no han sido recogidos durante un largo periodo de tiempo. También deben eliminarse los buzones de las cuentas borradas.
- Recomendar a los usuarios que no abran mensajes o anexos de mensajes que puedan contener virus informáticos.
- Asimismo, recomendar que no se envían mensajes que contengan una gran cantidad de

destinatarios. Para ello existen otro tipo de servicios, como las noticias.

Netscape Messenger como cliente POP3.

Messenger es el componente de **Netscape Communicator** encargado de gestionar los mensajes de correo electrónico. Sus características principales son:

- Permite recoger y enviar correo usando los protocolos POP3 e IMAP.
- El correo electrónico puede almacenarse en carpetas definidas por el usuario.
- Los mensajes pueden incluir anexos multimedia, páginas de hipertexto o cualquier otro tipo de archivos.
- El usuario puede incluir en sus mensajes firmas o etiquetas.
- Almacenar un libro de direcciones de correo electrónico.



Configuración de Netscape Messenger (1/2).

La configuración del programa se realiza en la opción "*Preferencias*" (*Preferences*) del menú "*Edición*" (*Edit*).

Como se muestra en el primer gráfico, la pantalla de configuración consta de una serie de menús. Entre ellos se encuentra el dedicado a establecer las características para correo y grupos de noticas ("*Mail & Groups*").

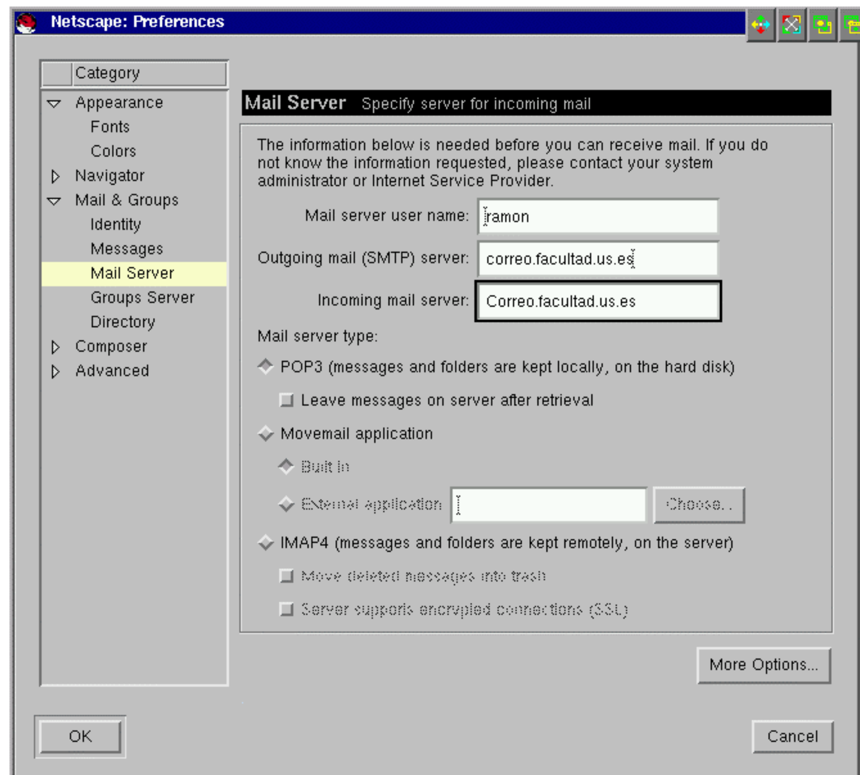
Las opciones relativas a la identidad de la persona ("*Identity*") permiten establecer las siguientes características:

- Nombre de la persona.
- Dirección de correo electrónico.
- Otra dirección de correo a la que se le envían las respuestas (no suele utilizarse).

- Organización.
- Fichero de texto con la firma del usuario.
- Indicador para incluir la tarjeta de visita del usuario.

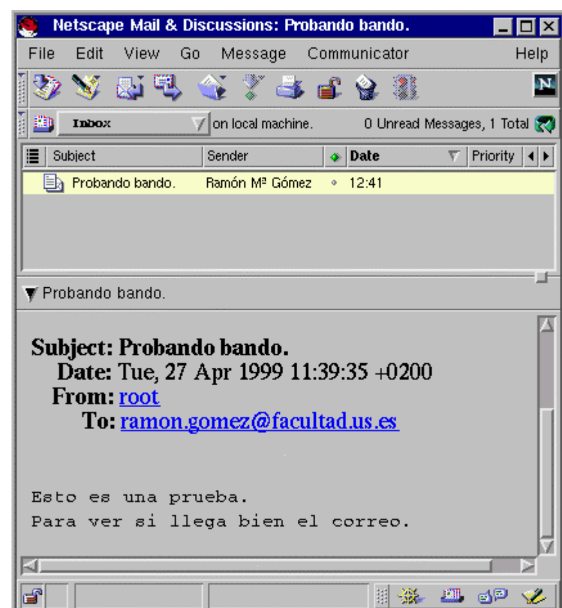
La segunda pantalla de configuración muestra las opcines relativas al servidor de correo electrónico. Éstas son:

- Nombre de usuario.
- Servidor SMTP de salida.
- Servidor de entrada, que suele coincidir con el servidor de salida.
- Elección del tipo de protocolo para la obtención del correo (POP3, IMAP4 o una aplicación específica).



Configuración de Netscape Messenger (2/2).

El último gráfico muestra un ejemplo de mensaje de correo leído con Netscape Messnger.



Netscape Messenger.

Conectividad entre Linux y la red de Microsoft.

La Red de Microsoft (MSN) nace con el Windows 3.11 para trabajo en grupo, con la idea de poder compartir unidades de discos e impresoras entre distintos ordenadores basados en este entorno. Las posteriores versiones de Windows (95, 98 y NT) han seguido utilizando esta técnica. El proceso de compartir recursos se basa en el protocolo **SMB** (*Session Message Block*), conocido también como **NetBIOS**, y es compatible con el protocolo **LanManager**.

Los usuarios pueden beneficiarse de la comodidad de manejo del entorno Windows para acceder a los datos de su cuenta o a las impresoras instaladas en el servidor Unix.

La utilidad SAMBA.

SAMBA es un conjunto de utilidades de dominio público –desarrollada principalmente por el programador australiano Andrew Tridgell– que implementa el protocolo SMB para ordenadores Unix, permientiéndoles funcionar como servidores de ficheros LanManager.

SAMBA realiza 4 funciones principales:

- Compartir datos Unix con ordenadores Windows (incluyendo cuentas de usuarios).
- Compartir unidades de discos Windows con ordenadores Unix.
- Compartir impresoras de Unix con máquinas Windows.
- Compartir impresoras de Windows con máquinas Unix.

Los componentes que forman SAMBA son:

`smbd`:	Provee los servicios de ficheros y de impresoras a los clientes SMB. Se configura mediante el fichero `smb.conf`.
`nmbd`:	Provee servicios de nombres NetBIOS (y WINS, opcionalmente) y soporte de visualización de datos.
`smbclient`:	Cliente SMB que permite acceder a otros servidores con apariencia similar al **`ftp`**.
`testparam`:	Permite comprobar el fichero de configuración `smb.conf`.
`smbstatus`:	Muestra los usuarios conectados al servidor.

Cuando un cliente realiza la petición de apertura de una sesión, se genera una copia del servidor **`smbd`**, que lee el fichero de configuración y posteriormente atiende todas las conexiones de dicho cliente. La copia del servidor termina cuando se cierran todas las conexiones para el cliente.

Los siguientes apartados van a describir cómo configurar correctamente un servidor SMB bajo Linux, cómo acceder desde clientes Windows al servidor y cómo acceder desde Linux a recursos que se encuentran en otros servidores SMB. Sin embargo, el administrador del sistema podrá encontrar mayor información sobre la configuración y el uso del paquete de programas SAMBA en las páginas de manuales suministradas con dicha utilidad o en la dirección de Internet http://samba.anu.edu.au/samba/.

Configuración previa.

Antes de configurar el servidpr SMB, el administrador del sistema debe realizar algunas comprobaciones previas.

- Verificar que el fichero `/etc/services` contenga una línea indicando el puerto de comunicaciones gestionado por el servidor **`smbd`**. El valor por omisión es el 139/TCP, como se muestra en el siguiente cueadro.

```
netbios-ssn     139/tcp         nbssn
```

- Si el ordenador va a ser utilizado también como servidor de nombres para el protocolo LanManager, el fichero `/etc/services` incluirá las líneas correspondientes a los puertos gestionados por el **`nmbd`**.

```
netbios-ns      137/tcp         nbns
netbios-ns      137/udp         nbns
netbios-dgm     138/tcp         nbdgm
netbios-dgm     138/udp         nbdgm
```

- Los procesos servidores de la aplicación pueden iniciarse de dos modos diferentes:

  - Como procesos independientes que se arrancan y se paren a través de los guiones reservados a tal efecto en los subdirectorios de `/etc/rc.d`. Debe comprobarse que exista el fichero correspondiente, que incluirá las órdenes siguientes:

    ```
    daemon smbd –D  OtrosParámetros
    daemon nmbd –D  OtrosParámetros
    ```

    Cuando se opta por este tipo de configuración no es necesario –aunque sí es recomendable– modificar el fichero `/etc/services`.

  - Como un servicio más del **`inetd`**. Deben incluirse las líneas correspondientes en el fichero `/etc/inetd.conf`, como se describe en el siguiente cuadro.

```
netbios-ssn stream tcp nowait root /usr/sbin/smbd OtrosParám
```

Conviene recordar que debe reiniciarse el servidor **`inetd`** siempre que se modifique el fichero `/etc/ined.conf`.

Los ficheros de configuración por omisión utilizados en ambos tipos de instalaciónes son:

`/etc/smb.conf`:	Configuración general del SMB.
`/etc/lmhosts`:	Configuración del servicio de nombres.

- Crear los directorios públicos y configurar las impresoras accesibles a través del SMB.

Configuración de SAMBA.

El archivo de configuración principal para la utilidad SAMBA es `smb.conf`, situado normalmente en el directorio `/etc`. Este es un fichero de texto editable únicamente por el usuario administrador (`root`), que consta de una serie de secciones y de parámetros.

Cada sección define un servicio prestado y los parámetros de dicha sección definen los atributos para dicho servicio.

Los nombres de las secciones vienen encerrados entre corchetes y cada uno de ellos consta de los parámetros definidos hasta la seguiente sección. Los parámetros tienen el formato: *`nombre`*`=`*`valor`*.

Existen 4 tipos de secciones:

`[global]:`	Consta de los parámetros aplicables por omisión a todos los servicios definidos.
`[homes]:`	Establece los parámetros para la conexión a las cuentas de los usuarios del servidor. Se utiliza el nombre de usuario como el nombre del servicio. **No permitir el acceso como invitado a este servicio.**
`[printers]:`	Define los parámetros para los servicios de impresión. Se utiliza el nombre de la impresora como nombre de servicio. **Este servicio debe tener activo el parámetro de impresión**.
Otras secciones:	Constan de un directorio de acceso, una serie de permisos y unas opciones extra. Pueden definirse servicios para usuarios invitados, que son los únicos que no requieren clave de autentificación.

La sustitución de variables permite asignar valores dinámicamente a los distintos parámetros. La siguiente table muestra algunas de las sustibuciones permitidas.

Formato	Descripción
`%S`	Nombre del servidor
`%P`	Directorio raíz del servicio.
`%u`	Nombre de usuario del servicio.
`%g`	Grupo primario del usuario `%u`.
`%H`	Directorio raíz de la cuenta del usuario `%u`.
`%v`	Versión de SAMBA.
`%h`	Nombre del ordenador servidor SAMBA.
`%L`	Nombre NetBIOS del servidor.
`%M`	Nombre del ordenador cleinte.
`%I`	Dirección IP del cliente.
`%m`	Nombre NetBIOS del cliente.
`%d`	Nº identificador del proceso gestor de un servicio (PID).
`%a`	Arquitectura de la máquina remota (samba, WfWg, WinNT o Win95).
`%T`	Fecha y hora.

La siguiente tabla muestra una descripción de los parámetros más importantes, clasificados en 2 grupos: parámetros que sólo pueden utilizarse en la sección `[global]` y parámetros generales.

Parámetros exclusivos de la sección `[global]`:	
`announce`	Establece el tipo de servidor (`NT`, `Win95` o `WfW`)
`announce version`	Versión del servidor.
`auto services`	Servicios que se añaden automáticamente a la lista de visualización.
`browse list`	Indica si el servidor muestra una lista de servicios. *Suele estar siempre activo.*
`case sensitive`	Realiza conversión entre mayúsculas y minúsculas.

`character set`	Establece el conjunto o mapa de caracteres.
`client code page`	Mapa de caracteres usado por el cliente.
`deadtime`	Minutos de inactividad antes de desconectar la sesión.
`default` `default service`	Servicio por omisión.
`default case`	Tipo utilizado cuando se desactiva `case sensitive`.
`domain logons`	Indica si SAMBA se convierte en servidor de dominio para Win95.
`domain master`	Permite que el servidor obtenga datos de toda la red o sólo de su red local.
`hosts equiv`	Establece el fichero con usuarios y ordenadores que no requieren clave de acceso. *Añadir sólo los administradores y sus máquinas.*
`load printers`	Permite cargar inicialmente todas las impresoras definidas.
`local master`	Permite que el **`nmbd`** se convierta en servidor de visualización para la red local.
`lock directory`	Directorio donde se almacenan los ficheros de bloqueo.
`log file`	Establece el fichero con el registro histórico de incidencias.
`name resolve order`	Orden para los tipos de resolución de nombres.
`netbios aliases`	Otros nombres NetBIOS para el servidor.
`netbios name`	Nombre NetBIOS del servidor.
`networkstation user login`	**Recomendado para aumentar la seguridad del servidor.**
`null passwords`	Permite el acceso mediante claves nulas. *Poco recomentado.*
`password program`	Establece el programa para cambiar claves de usaurios. **Usar el suministrado por el sistema operativo (`/bin/passwd`).**
`preferred master`	Indica que SAMBA es servidor preferente del grupo de trabajo.
`preserve case`	Indica si los nuevos ficheros se crean con los caracteres provistos por el cliente o con el tipo establecido en `default case`.

`root`
`root dir`
`root directory`
	Directorio raíz del servidor. *Añade más seguridad, pero impide acceder a zonas del árbol de directorios.*

`security`	Modo de seguridad de SAMBA. **Establecer `security=user` en los servidores Linux con cuentas de usuarios.**
`status`	Permite obtener un informe de las conexiones activas.

`time server`	El **`nmbd`** actúa como servidor de tiempo para clientes Windows.
`unix real names`	SAMBA incluye los nombres reales de los usuarios UNIX.
`username map`	Indica el fichero con la conversión de usuarios entre clientes y servidor.
`workgroup`	Indica el grupo de trabajo Windows.
Parámetros que pueden incluirse en cualquier sección:	
`admin users`	Establece los usuarios administradores del servicio. **Incluir sólo los operadores de la máquina y el usurario `root`.**
`available`	Permite desactivar servicios.
`browseable`	El servicio se muestra en la lista de servicios.
`comment`	Descripción del servicio.
`clone`	Permite *clonar* los parámetros de otro servicio.
`create mask` `create mode`	Máscara AND para la creación de ficheros.
`directory mask` `directory mode`	Máscara AND para la creación de directorios.
`dont descend`	Evita que el cliente pueda entrar en ciertas partes del árbol de directorios. **Incluir, al menos, los directorios `/proc` y `/dev`.**
`fake oplock`	Si se activa, varios usuarios no pueden acceder a la vez al mismo fichero.
`follow simlinks`	Permite que el SAMBA siga los enlaces simbólicos.
`force create mode`	Los archivos se crean con los permisos especificados.
`force directory mode`	Los directorios se crean con los permisos especificados.
`force group`	Grupo para todas las conexiones al servicio.
`force user`	Usuario para todas las conexiones al servicio.
`guest account`	Usuario que accede al servicio como invitado.
`guest ok` `public`	El cliente no necesita clave para acceder al servicio.
`guest only`	Sólo permite acceso como invitado.
`hide files`	Establece los ficheros y directorios accesibles pero invisibles.
`hosts allow` `allos hosts`	Conjunto de máquinas y/o dominios con acceso al servicio.
`hosts deny` `deny hosts`	Conjunto de máquinas y/o dominios sin acceso al servicio.

`invalid users`	Lista de usuarios sin permiso de conexión al servicio.
`locking`	Permite atender peticiones de bloqueo.
`lppause command`	Mandato que permite parar trabajos de impresión.
`lpq command`	Mandato para obtener información de los trabajos de impresión.
`lpresume command`	Mandato para restaurar trabajos de impresión.
`lprm command`	Mandato para borrar trabajos de impresión.
`mangled map`	Establece conversión adecuada para ficheros DOS (8.3).
`mangled names`	Permite convertir nombres largos en nobres DOS (8.3).
`max connections`	Nº máximo de conexiones a un servicio.
`max log size`	Tamaño máximo en KB del fichero hostórico.
`min print space`	Espacio mínimo de disco (en KB) para que el usuario pueda mandar trabajos a la cola de impresión.
`path` `directory`	Directorio para el usuario conectado al servicio.
`postexec`	Mandato ejecutado en la desconexión del servicio.
`preexec` `exec`	Mandato ejecutado en la conexión al servicio.
`print command`	Mandato usado para procesar los trabajos en cola de impresión.
`printable` `print ok`	Los clientes pueden mandar trabajos a la cola de impresión.
`printer` `printer name`	Nombre de la impresora.
`read list`	Usuarios con permiso sólo de lectura en el servicio.
`root postexec`	Como `postexec`, pero ejeutado por `root`. *Por ejemplo, se usa para desmontar un CD-ROM.*
`root preexec`	Como `preexec`, pero ejecutado por `root`. *Por ejemplo, para montar un CD-ROM.*
`valid users`	Lista de usuarios y grupos con permiso de conexión al servicio.
`veto files`	Lista de ficheros y directorios vetados (ni visibles, ni accesibles).
`volume`	Establece etiquetas de volúmenes (usado en algunos CD-ROMs).
`writable` `write ok`	Los usuarios del servicio pueden crear o modificar archivos.
`write list`	Lista de los únicos usuarios con permiso de escritura.

Seguridad.

El administrador del servidor SMB deberá tener en cuenta las siguientes recomendaciones para controlar el acceso a los datos suministrados.

- Por seguridad, ningún fichero ejecutable de la utilidad SAMBA debe tener activo el permiso SUID.
- Los archivos que contienen el registro histórico de incidencias deben incluirse en un directorio con acceso restringido al administrador (usuario `root`).
- Únicamente `root` debe tener permiso de escritura en el fichero de configuración `smb.conf`. No es necesario que el resto de usuarios tengan permiso de lectura sobre dicho fichero.
- Si se arrancan los procesos SAMBA como meta-servicios del **`inetd`**, puede utilizarse el programa TCP-Wrappers para aumentar las prestaciones de seguridad.
- Tener extremo cuidado con los parámetros globales y por omisión del servidor, porque pueden activar permisos de acceso no deseados para algunos servicios.
- El usuario nunca deberá activar en Windows95 la opción "*Guardar contraseña en la lista de contraseñas*" porque otro usuario del mismo PC podría capturar su clave fácilmente.

Ejemplo.

La siguiente tabla muestra los ficheros de configuración del servidor Samba creado para el ejemplo. Posteriormente se presentarán unos gráficos que muestran el acceso al servidor desde un ordenador personal con Windows95. En último lugar se muestra el acceso al servidor desde un servidor Linux mediante la utilidad **`smbclient`**.

smb.conf	
`[global]` `  allow hosts = *.facultad.us.es`	Ordenadores con acceso al servidor SMB.
`  workgroup = proyecto`	Grupo de trabajo para Windows.
`  comment = Servidor Samba de mi Factuldad`	Descripción del servidor.
`  dont descend = /proc,/dev`	Directorios que aparecen vacíos.

`  printing = bsd`
`  printcap name = /etc/printcap`
`  load printers = yes`
	Órdenes básicas para impresión.

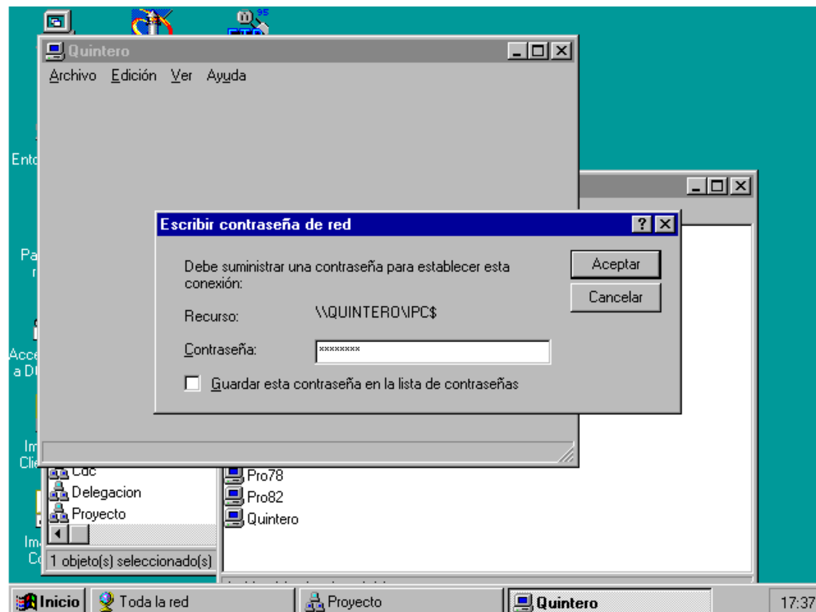
`  log level = 1` `  log file = /var/log/samba/samba-log.%m`	Fichero histórico de incidencias.

short preserve case = yes preserve case = yes	Se conservan los nonbres de los nuevos archivos.
lock directory = /var/lock/samba locking = yes strict locking = yes	Características de los bloqueos para accesos compartidos a los ficheros.
security = user invalid users=root	Los usuarios se autentifican por su clave Unix (por seguridad se prohibe el acceso a `root`).
[homes] comment = Directorios de alumnos path = %H only users = yes read only = no preserve case = yes short preserve case = yes create mask = 0700 directory mask = 0700	Incluye las carácterísticas de acceso a las cuentas de usuarios, estableciendo: descripción, directorio de la cuenta, no se permite aceso público, se activa el permiso de escritura y las máscaras para creación de nuevos archivos y directorios.
[printers] comment = Impresoras path = /var/spool/samba browseable = no printable = yes public = no writable = no create mode = 0700	Características de las impresoras definidas, incluyendo: descripción, directorio donde se almacenan los trabajos y características de visualización, acceso y creación.
[cdrom] comment = CDROM path = /mnt/cdrom root preexec = /bin/mount /mnt/cdrom root postexec = /bin/umount /mnt/cdrom read only = yes browseable = yes	Características de los CD-ROMs montados en el servidor, tales como: descripción, punto de montaje, órdenes para montaje y desmontaje de discos y características de acceso.
[aplicaciones] comment = Aplicaciones Windows path = /home/aplic read only = yes writable = yes write list = instal,ramon create mask = 0764 directory mask = 0775 force directory mode = 2000 public = no browseable = yes	Servicio de aplicaciones Windows almacenadas en el servidor SMB. Se establecen los parámetros: descripción, acceso de sólo lectura global y de escritura para los usuarios `instal` y `ramon`, máscaras de creación de archivos y directorios y no se permite el acceso anónimo.
smbusers	

# Unix_name = SMB_name1 SMB_name2 ... 	
root = administrador admin
nobody = guest pcguest smbguest
	Establece correspondencias entre usuarios SMB y usuarios Unix.

Para acceder desde Windows95, el usuario deberá seguir los siguientes pasos:

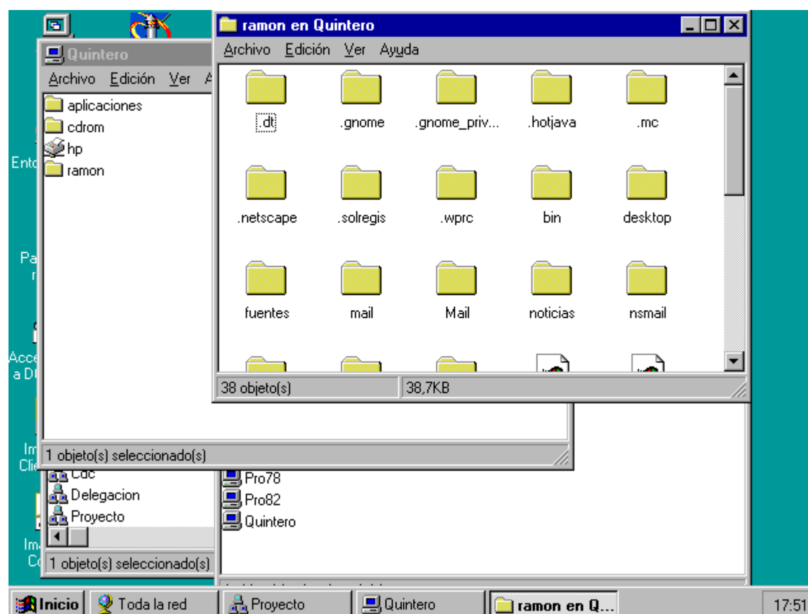
1. Entrar en el sistema utilizando su nombre de usuario –o su alias–en Samba y con una clave cualquiera.



Acceso desde Windows95.

2. Para realizar la conexión deberá buscar el nombre del servidor en el “*Entorno de Red*” y en el grupo de trabajo especificado (en el ejemplo “*Proyecto*”).

3. El servidor utilizará el nombre de usuario establecido en el arranque y pedirá la clave (en el ejemplo, la clave corresponderá con la del usuario en el servidor Unix).



Conexión a la cuenta del usuario.

4. **Por seguridad, desactivar siempre la opción “*Guardar contraseña en la lista de contraseñas*”.**

5. Cuando el usuario entra en el servidor, éste le muestra los distintos servicios activos.

6. Si la persona elige el servicio correspondiente con su nombre de usuario Unix, podrá acceder a los datos almacenados en su cuenta.

Protección del servidor.

Una de las funciones principales del gestor de un servidor conectado a la red es establecer restricciones de acceso para cada uno de los servicios ofrecidos, que eviten o prevengan la posibilidad de piratear, dañar o acceder ilegalmente a la información de los usuarios o a las aplicaciones instaladas en él.

Ya se han comentado algunas de las prestaciones de seguridad –aquellas que vienen incluidas o que mejoran cada servicio de red–, sin embargo, también existe una serie de utilidades genéricas que establecen reglas comunes de seguridad, tanto para el servidor, como para el resto de ordenadores de su subred.

En este capítulo se describirán brevemente algunos de los métodos para aislar y proteger servidores y subredes. Si el lector quiere obtener una mayor información sobre seguridad en Unix o sobre las aplicaciones aquí descritas, deberá acudir a las páginas de manual correspondientes, a la documentación incluida con la distribución del sistema operativo o a las direcciones de Internet especializadas en el tema.

Seguridad con TCP-Wrappers.

TCP-Wrappers es una herramienta de dominio público –incluida en las distribuciones del Linux de RedHat– que permite al administrador del sistema establecer las reglas principales de acceso a los servicios activados mediante el **`inetd`**.

El modo de operar de TCP-Wrappers es el siguiente:

1. El **`inetd`** activa un proceso de TCP-Wrapper cuando se realiza una petición de un servicio protegido con dicha aplicación.
2. TCP-Wrappers procesa sus ficheros de configuración y comprueba si el ordenador cliente tiene permiso de acceso para el servicio especificado.
   a) Si no se permite el acceso a ese servicio desde ese determinado ordenador, se ejecutan las instrucciones establecidas a tal efecto por el administrador –como registrar el evento en un fichero histórico o mandar correo electrónico al gestor del sistema– y se corta la conexión.
   b) Si el cliente tiene permiso de acceso al servicio desde dicho ordenador, se ejecutan las operaciones pertinentes –como registrar el evento en un fichero– y se lanza automáticamente el proceso que gestiona el servicio correspondiente.

La siguiente tabla describe los 2 ficheros principales que configuran las reglas de acceso (normalmente incluidos en el directorio `/etc`).

hosts.allow	
Descripción:	Establece los ordenadores con de acceso a determinados servicios.
Formato:	`Servicios : Ordenadores : [ Órdenes ]`

Ejemplo:	`ALL: consola.facultad.us.es`
	`in.telnetd ipop3d: .facultad.es`
	`in.ftpd: ALL`

hosts.deny	
Descripción:	Establece los ordenadores que NO pueden acceder a cada servicio.
Formato:	`Servicios : Ordenadores : [ Órdenes ]`
Ejemplo:	`ALL: ALL`

Una vez establecidos los requisitos de acceso, el administrador de la máquina estimará los procesos que pasarán por el filtro de TCP-Wrappers. Para ello deberán modificarse las líneas correspondientes del archivo `/etc/inetd.conf`, realizando las siguientes sustituciones:

- Indicar en el campo "*Servidor*" el camino completo para el ejecutable de TCP-Wrappers.
- Establecer en el campo "*Argumentos*" el mandato y los argumentos del proceso servidor coreespondiente.

Antes:

```
...
ftp    stream tcp nowait root /usr/sbin/in.ftpd in.ftpd -l –a
telnet stream tcp nowait root /usr/sbin/in.telnetd in.telnetd
pop-3  stream tcp nowait root /usr/sbin/ipop3d ipop3d
...
```

Después:

```
...
ftp    stream tcp nowait root /usr/sbin/tcpd  in.ftpd -l –a
telnet stream tcp nowait root /usr/sbin/tcpd  in.telnetd
pop-3  stream tcp nowait root /usr/sbin/tcpd  ipop3d
...
```

Servidor representante.

Un servidor **representante** –denominado también **representante oculto**, *proxy* o *proxy caché*– es un ordenador que realiza las peticiones en nombre de sus clientes. A grandes resgos, el proceso de delegación de servicio es el siguiente:

- Un usuario situado en una máquina de la red oculta quiere realizar una determinada petición a un servidor de Internet.
- Su programa *navegador* o cliente pide los datos al servidor representante de la subred.

- Si éste tiene almacenados los datos deseados, los transfiere directamente al ordenador del usuario.

- En caso contrario, transfiere la petic ión al servidor origen, captura los resultados, los envía al usuario y los almacena, previendo la posibilidad de un acceso posterior a la misma información.

- Si se satura el espacio reservado por el representante, éste eliminará los datos menos utilizados.

El ordenador donde se ejecuta el representante debe estar equipado con uno o varios discos duros de gran capacidad –un minimo de 2 GB para *caché*–, que permitan guardar la información capturada, evitando realizar accesos innecesarios a la máquina de origen. Otro factor que mejora las prestaciones es la inclusión de una tarjeta de red rápida (100-BaseT). Sin embargo, estos factores estarán en función del número de accesos previstos y de la cantidad de información que desea ser recogida por los usuarios del servidor.

Al igual que en el resto de apartados, ilustaremos con ejemplos las características de configuración para un programa servidor representante. En este caso hemos elegido la utilidad **Squid versión 2**, de libre disposición, pero no incluida en la distribución del Linux de RedHat.

El administrador del sistema deberá importar el programa –consultar la página principal de Squid en http://squid.nlanr.net/ o su página de documentación en http://cache.is.co.za/squid/–, compilarlo según las instrucciones incluidas con la distribución e instalar los archivos ejecutables y de configuración (dicho proceso utiliza por omisión el directorio `/usr/local/squid`).

La distribución de Squid incluye una serie de componentes –o las direcciones correspondientes para descargarlos– que permiten ampliar las capacidades del servidor (acceso a correo vía POP3, autentificación a través de SAMBA, etc.).

El archivo de configuración `squid.conf` –qie se encuentra normalmente en el directorio `/usr/local/squid/etc`– está subdividido en varias secciones, donde cada una de ellas consta de varias instrucciones. La siguiente tabla muestra los formatos de algunas de las opciones más utilizadas.

Opciones de red:	
`http_port` *`Puerto`*	Puerto donde se atienden las peticiones HTTP de los clientes.
`icp_port` *`Puerto`* `thcp_port` *`Puerto`*	Puertos de comunicación con otros representantes.
Opciones para representantes vecinos:	
`cache_peer` *`Tipo PuertoHTTP PuertoICP`* `[`*`Opciones`*`]`	Características para la conexión a un representante vecino.
`cache_peer_domain` *`Proxy Dominios`*	Accede a datos de ciertos dominios guardados en otro *proxy*.
`icp_query_timeout` *`MiliSeg.`*	Tiempo máximo de espera para peticiones ICP.
`dead_peer_timeout` *`Seg.`*	Tiempo para considerar que se ha perdido la comunicación.

`hierarchy_stoplist` *Peticiones*	Peticiones atendidas directamente, sin atender a los vecinos.
`no_cache` *ListaControlAcceso*	Objetos que nunca serán capturados por el representante.
Opciones para el tamaño de la memoria:	
`cache_mem` *Tamaño*	Cantidad de memoria usada para el tráfico de información.
`maximum_object_size` *Tamaño*	Tamaño máximo que puede tener un objeto almacenado.
`fqdncache_size` *Número*	Nº máximo de entradas capturadas en el servicio de nombres.
Opciones para archivos y directorios:	
`cache_dir` *Directorio MBytes Nivel1 Nivel2*	Localización de un directorio de *caché*, tamaño de almacenamiento y nº de subdirectorio de 1er y 2º nivel.
`cache_access_log` *Fichero*	Fichero histórico de accesos de los clientes.
`cache_log` *Fichero* `cache_store_log` *Fichero* `cache_swap_log` *Fichero*	Históricos que guardan incidencias del servidor (accesos, almacenamiento y paginación de memoria).
`pid_filename` *Fichero*	Fichero que contiene el identificador de proceso del servidor.
`debug_options` *Sección,Nivel*	Opciones para depuración (formato utilizado por **`syslogd`**).
`client_netmask` *Máscara*	Máscara de red para los clientes
Opciones para otros programas:	
`ftp_user` *ClaveUsuarioAnónimo*	Clave para accesos a FTP anónimos.
`cache_dns_program` *Mandato* `dns_children` *NºProcesos*	Programa y nº de hijos para buscar en el servicio de nombres.
`unlinkd_program` *Mandato*	Programa para borrado de objetos.
`authentificate_program` *Mandato* `authentificate_children` *Procesos*	Programa para autentificación y nº de procesos hijos.
Opciones para ajustar el servidor:	
`request_size` *Tamaño* `KB`	Tamaño máximo para cada petición.
`refresh_patern [-i]` *Expresión MínMinut Porcent MáxMinut* [*Opc*]	Características del algoritmo empleado para refrescar los datos almacenados.
`refresh_age` *Tiempo*	Tiempo máximo que permanece un objeto sin ser recargado.
`positive_dns_ttl` *Tiempo* `negative_dns_ttl` *Tiempo*	Validez para las peticiones correctas y erróneas al DNS.
`connect_timeout` *Tiempo* `read_timeout` *Tiempo* `request_timeout` *Tiempo*	Tiempos para el establecimiento de conexión, lectura, petición.
`client_lifetime` *Tiempo* `shutdown_lifetime` *Tiempo*	Tiempos que un cliente puede quedar conectado (en estado normal y durante la parada del servidor).

Opciones para listas de control de accesos:	
acl *Nombre Tipo Expresión*	Define una lista de control de acceso.
http_access allow|deny [!]*Listas* icp_access allow|deny [!]*Listas* miss_access allow|deny [!]*Listas*	Permite o niega el accesos HTTP, ICP o erróneos a una o varias listas de control de acceso.
cache_access allow|dwny [!]*ACLs*	Permite o niega el acceso desde otros representantes.
Opciones de administración:	
cache_mgr *DirecciónCorreo*	Dirección de correo electrónico del administrador del servidor.
cache_effective_user *Usuario* cache_effective_group *Grupo*	Usuario y grupo reales utilizados por procesos de **Squid**.
visible_hostname *NombreMáquina*	Alias usado para el representante.
Opciones varias:	
logfile_rotate *Número*	Nº Máximo de ficheros históricos utilizados.
append_domain *Dominio*	Domino por omisión.
deny_info *PágError Lista*	Mensaje de error para peticiones fallidas de una lista.
forwarded_for on|off	Permite incluir el nombre o la dirección IP en peticiones HTTP.
cachemgr_passwd *Clave Acciones*	Establece claves para cada acción administrativa.
always_direct allow|deny [!]*ACLs* never_direct allow|deny [!]*ACLs*	Permite o inhibe que las peticiones de las listas siempre o nunca se dirijan al servidor origen.
icon_directory *Directorio* error_directory *Directorio*	Directorios de iconos y de mensajes de errores.
snmp_port *Puerto*	Puerto utilizado para gestión mediante el protocolo SNMP.
coredump_dir *Directorio*	Directorio para ficheros de volcado de procesos erróneos.

El siguiente ejemplo muestra el formato utilizado en la creación de listas de control de accesos.

acl servlocales dstdomain facultad.us.es always_direct allow servloclaes	Todas las peticiones se redirigen a servidores locales.
acl todos src 0.0.0.0/0.0.0.0 icp_access allow todos	Permite acceso ICP desde cualquier representante.
acl PrtsSeguros port 80 21 443 563 70 210 1025-65535 http_access deny !PrtsSeguros	Sólo permite acceder a los puertos considerados seguros.
acl PuertosSSL port 443 563 acl CONNECT method CONNECT http_access deny CONNECT !PuertosSSL	Sólo permite usar el método CONNECT en puertos SSL.

```
acl RedInterna src 192.168.2.0/255.255.255.0
acl todos src 0.0.0.0/0.0.0.0
http_access allow RedInterna
http_access deny todos
```

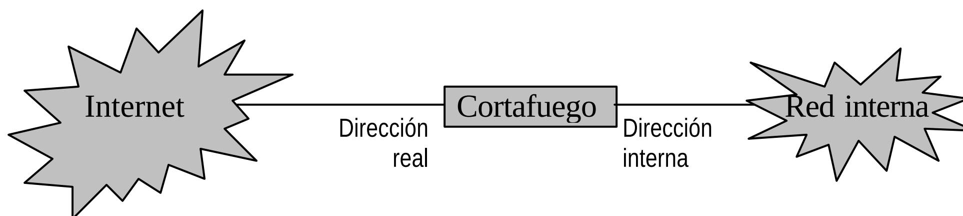
Sólo se permite el acceso desde ordenadores de una subred interna 192.168.2.x.

Cortafuego.

Un **Cortafuego** –conocido también por la palabra inglesa *firewall*– es un servidor que protege una subred interna (con direcciones 192.168.x.x) de accesos no autorizados realizados desde Internet, realizando las funciones de redirección y filtrado para los paquetes de datos entre ambas redes.

Para ello, el ordenador deberá contar con 2 tarjetas de red:

- una de ellas conectada a Internet, con dirección real y que realizará las funciones de *encaminador* de paquetes para el otro adaptador
- y la otra conectada a la subred, con dirección interna y que actuará como *encaminador* de dicha subred.



Estos servidores deben contar con un alto grado de seguridad, para evitar que los intrusos lo utilicen como punto de acceso a la red interna. Existen 2 filosofías fundamentales para instalar un cortafuego:

1. *Se permite todo aquello que no está expresamente prohibido.* Esta configuración deja una mayor libertad de acción a los usuarios, por lo tanto el administrador deberá prever los posibles ataques que pueden ver peligrar la seguridad del servidor y las acciones a tomar en cada caso.
2. *Se prohibe todo aquello que no está expresamente permitido.* Por el contrario, esta otra configuración ofrece mayor seguridad, puesto que el servidor sólo permitirá acceder a aquellos servicios verificados por el administrador y, por lo tanto, considerados como seguros.

El gestor de la máquina (usuario `root`) deberá configurar correctamente los 2 dispositivos de red y preparar el sistema operativo para que éste pueda utilizarlos, recompilando el núcleo central (o *kernel*).

La siguiente tabla muestra las características más recomendadas para adecuada configuración.

En “configuración general”:	Activar soporte de red.
En “opciones de red”:	Activar cortafuego de red. Activar red TCP/IP. Activar redirección IP sólo si se desea filtrado IP. Activar cortafuego IP. Desactivar enmascaramiento IP. Activar contabilidad IP. Desactivar túnel IP. Desactivar alias IP. Desactivar modo compatible PC/TCP. Desactivar ARP inverso.
En “dispositivo de red”:	Activar soporte para dispositivo de red. Activar Ethernet a 10 Mbps o a 100 Mbps. Seleccionar la tarjeta de red.

Una vez compilado el nuevo núcleo, deberá configurarse el programa **LILO** (cargador de Linux), añadiendo los parámetros físicos de cada tarjeta en el archivo `/etc/lilo.conf`.

Por último, deben establecerse las direcciones de red para cada tarjeta. Los archivos de configuración tratados en la siguiente tabla, se encuentran normalmente en el directorio `/etc/sysconfig/network-scripts`.

ifcfg-eth0	
DEVICE=eth0 IPADDR=*DirecciónIP* NETMASK=255.255.255.0 NETWORK=*DirecciónDeRed*.0 BROADCAST=*DirecciónDeRed*.255 GATEWAY=*DirecciónIPDelEncaminador* ONBOOT=yes	La tarjeta conectada a Internet hará de encaminador para la 2ª tarjeta de red.
ifcfg-eth1	
DEVICE=eth1 IPADDR=*DirecciónIPInterna* NETMASK=255.255.255.0 NETWORK=*DirecciónDeRedInterna*.0 BROADCAST=*DirecciónDeRedInterna*.255 GATEWAY=*DirecciónIPTarjetaEth0* ONBOOT=yes	El 2º dispositivo tendrá la dirección reservada al encaminador de la subred interna (por ej.: **192.168.2.1**).

Uno de los programas cortafuegos de libre distribución más utilizados es **Firewall Toolkit (FWTK)** de Trusted Information Systems (empresa adquirida recientemente por Network Associates). Esta utilidad no viene incluida con la distribución de RedHat 5.2, así que debe ser importada –de ftp://ftp.tis.com– o solicitada por correo electrónico –a fwtk-request@tis.com–, compilada e instalada por el administrador del servidor (para ello, seguir las instrucciones incluidas con el programa).

El proceso de configuración de **FWTK** consta de los siguientes pasos:

1. **Añadir o modificar la lista de servicios protegidos** en el archivo `/etc/services`.

```
ftp-gw     21/tcp          # Nuevo servicio FTP
telnet-a   24/tcp          # Telnet para administrador
ssl-gw    443/tcp          # Servicio SSL
...
authsrv  7777/tcp ident   # Autentificación de usuarios
```

   El servicio de verificación de usuarios (**`authsrv`**) se utiliza para establecer las prestaciones de seguridad para el resto de utilidades.

   Es importante destacar que el servicio **`telnet`** *real* al ordenador del cortafuego se realiza por un puerto distinto al 23/TCP –este puerto se usa para filtrar **`telnet`** de la red interna– y el administrador será el único usuario que podrá acceder (y sólo desde máquinas seguras).

2. **Modificar las entradas de los programas gestores de servicios** en el archivo `/etc/inetd.conf`.

```
...
# Cortafuego para FTP.
ftp-gw   stream tcp nowait root /usr/local/etc/ftp-gw  ftp-gw
# Cortafuego para Telnet.
telnet   stream tcp nowait root /usr/local/etc/tn-gw   tn-gw
# Telnet local para el administrador.
telnet-a stream tcp nowait root /usr/local/etc/netacl  in.telnetd
# Cortafuego para WWW.
http-gw  stream tcp nowait root /usr/local/etc/http-gw http-gw
# Cortafuego para SSL (comunicación segura).
ssl-gw   stream tcp nowait root /usr/local/etc/ssl-gw  ssl-gw
# Servicio de autentificación.
authsrv  stream tcp nowait root /usr/local/etc/authsrv authsrv
...
```

   Las utilidades también permiten el filtrado de otros servicios como el correo electrónico, el acceso delegado a noticias o a la aplicación Gopher, etc.

3. **Configurar la tabla de permisos de acceo a cada servicio**, modificando el fichero `netperm-table`, localizado en el directorio de instalación de **FWTK** (normalmente `/usr/local/etc`). Este archivo consta de una sección para cada servicio y cada una de ellas incluye directivas que configuran las características de acceso para dicho servicio.

   Ejemplo:

Directicas para `ftp-gw`:	

`ftp-gw: denial-msg   /usr/local/etc/ftp-deny.txt`
`ftp-gw: welcome-msg  /usr/local/etc/ftp-welcome.txt`
`ftp-gw: help-msg     /usr/local/etc/ftp-help.txt`
`ftp-gw: permit-hosts 150.214.69.100`
`ftp-gw: permit-hosts 192.168.2.* -log { retr stor } -auth { stor }`
`ftp-gw: permit-hosts * –authall`
`ftp-gw: timeout      3600`
`# En el ejemplo, el PC del administrador es el`
`# 150.214.69.100 y la red oculta, la 192.168.2.X.`
	Establece los mensajes para denegación, bienvenida y ayuda, indica los ordenadores con permiso de acceso y cuáles requieren almacenamiento histórico y/o autentificación y establece el tiempo máximo de conexión.

<table>
<tr><th colspan="2">Directicas para <code>tn-gw</code>:</th></tr>
<tr><td><pre>tn-gw: denial-msg   /usr/local/etc/tn-deny.txt
tn-gw: welcome-msg /usr/local/etc/tn-welcome.txt
tn-gw: help-msg   /usr/local/etc/tn-help.txt
tn-gw: permit-hosts 192.168.2.* -passok -xok
tn-gw: permit-hosts * -auth
tn-gw: timeout   90</pre></td><td>Establece los mensajes para denegación, bienvenida y ayuda, indica los ordenadores con permiso de acceso y cuáles tienen permiso para autentificación y/o cambio de claves y el tiempo máximo para establecer la conexión.</td></tr>
<tr><th colspan="2">Directicas para <code>netacl-in.telnetd</code>:</th></tr>
<tr><td><pre>netacl-in.telnetd: permit-hosts 150.214.69.100 -exec
/usr/sbin/in.telnetd</pre></td><td>Sólo el administrador debe ejecutar <code>telnet</code> real desde ordenadores seguros.</td></tr>
<tr><th colspan="2">Directicas para <code>http-gw</code>:</th></tr>
<tr><td><pre>http-gw: userid   root
http-gw: directory   /home/privado
http-gw: default-httpd www.facultad.us.es
http-gw: hosts   192.168.2.* -log { read write
ftp }
http-gw: deny-hosts   *
http-gw: timeout   90</pre></td><td>Indica el usuario y directorio donde se almacenan archivos FTP de entrada, el servidor HTTP por omisión, establece los permisos para los ordenadores de la red oculta, niega el acceso al resto y activa el temporizador de conexión.</td></tr>
<tr><th colspan="2">Directicas para <code>ssl-gw</code>:</th></tr>
<tr><td><pre>ssl-gw: hosts   192.168.2.* -dest { !127.0.0.*
!150.214.69.* *:443:563 }
ssl-gw: deny-hosts *
ssl-gw: timeout   300</pre></td><td>Permite únicamente a los ordenadores de la red interna el acceso SSL delegado a ordenadores de Internet, excepto el servidor y los ordenadores de la red externa e indica el tiempo de establecimiento de conexión.</td></tr>
<tr><th colspan="2">Directivas para <code>authsrv</code>:</th></tr>
<tr><td><pre>authsrv: database   /usr/local/etc/fw-authdb
authsrv: permit-hosts 127.0.0.1
authsrv: badsleep   1200
authsrv: nobogus   true

*:   authserver   127.0.0.1 7777</pre></td><td>Establece el camino de la base de datos de usuarios, los clientes permitidos y el tiempo de espera y muestra mensajes de error.

Todos los servicios usarán el autentificador.</td></tr>
</table>

4. **Establecer las cualidades para los usuarios incluidos en el servicio de autentificación.** La utilidad **authsrv** incluye una serie de órdenes que gestionan la base de datos donde se almacenan las características de los usuarios y grupos fictios que serán autentificados antes de ejecutar un determinado servicio.

   Ejemplo:

```
# authsrv
-administrator mode-
authsrv# adduser admin 'Admin BD Auth'
ok - user added initially disabled
authsrv# ena admin
enabled
authsrv# superwiz admin
set wizard
authsrv# proto admin pass
changed
authsrv# pass admin “clave”
Password changed
authsrv# list
Report for users in database
user        group       longname        ok?  proto       last
----        -----       --------        ---  -----       ----
admin                   Admin BD Auth   ena  pass        never
authsrv# quit
#
```

El lector podrá obtener mayor información sobre los programas cortafuegos –y sobre las utilidades FWTK– en la documentación incluida con el sistema operativo o en la dirección http://www.tis.com/.